Master's Thesis

Using 3D Imaging Radar for Indoor Localization and Mapping

Ruoyang Xu
CMU-RI-TR-22-37
August 2022

Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Prof. Michael Kaess, Chair
Prof. Sebastian Scherer
Jay Patrikar

Submitted in partial fulfillment of the requirements
for the degree of Master of Science.

Copyright © 2022 Ruoyang Xu

Keywords: 3D Imaging Radar; Range Sensing; Visual Learning; SLAM

Abstract

3D Imaging Radars offer robust perception capability through visually demanding environments due to the unique penetrative and reflective properties of millimeter waves. However, the utilization of Imaging radar for robot navigation and mapping remains under-explored due to the complex data representation and extremely noisy measurements. Current approaches for 3D perception with imaging radar require knowledge of environment geometry, accumulation of data from multiple frames for perception, or accurate between-frame motion.

This thesis makes contributions in the domains of utilizing 3D Imaging radar for robot navigation and mapping. We propose a learning-based method to regress radar measurements in the form of cylindrical depth maps using LiDAR supervision. Due to the limitation of the regression formulation, directions where the radar beam could not reach will still generate a valid depth. Our method additionally learns a 3D filter to remove those pixels. Experiment results show that our system generates visually accurate depth estimation.

We confirm the overall effectiveness of this learned frontend by applying it to common downstream robotics tasks. We show that it is possible to use the learned depth map to retrieve doppler velocity measurements and infer a sensible radar-frame velocity. Applying the depth map to probabilistic occupancy mapping with ground truth trajectory generates point cloud maps that are visually consistent with LiDAR maps. Lastly, we show that by explicitly looking for large 3D planes in the learned depth map, and modeling structural constraints, it is possible to perform indoor SLAM with a noisy odometry source.

Acknowledgments

I first would like to thank my advisor Dr. Michael Kaess. He believed in my potential when I had little to no background experience in this research area, and took me on as an MS student. I have benefited from his experience and knowledge as a robotics researcher and practitioner. Over the past two years, Michael gave me the freedom that I doubt I would ever see in a MS student again, and allowed me to freely learn and explore. I will always be grateful for his guidance, support, and the unbelievable patience. I am thankful to Dr Sebastian Scherer for serving on my committee, and Jay Patrikar, also for the many discussions we went through.

I want to thank all my friends in the robot perception lab. Wei, Akshay, Sam, Akash, Mohamad, Alex, Yehonathan, Dan, Allison, Allie, Monty, Suddu, and Paloma. You made the experience in the office for me, and none of this would have been possible without you. In addition, I would like to thank all my friends and colleagues in the Master of Science in Robotics cohort of '22. Life was more exciting when I got to learn about research outside of my own fields. I also owe thanks to my friends who helped me to make this pandemic more bearable: Yilun, Satoru, Ian, Zeyuan, and many others. Lastly, I would like to thank my parents for their continued trust, faith, and utmost support.

I further acknowledge funding support from Amazon Lab 126 for this work.

Contents

1 Introduction **1**
1.1 Motivation 1
1.2 Scope and Approach 2
1.3 Organization and Contributions 3

2 Background **5**
2.1 Introduction to SLAM 5
2.1.1 SLAM and Factor Graphs 5
2.1.2 Frontend of SLAM 7
2.2 mmWave FMCW Radar 8
2.2.1 Introduction to mmWave FMCW Radar 8
2.2.2 Imaging Radar using Cascaded FMCW mmWave Radar Systems in MIMO Configuration 11
2.2.3 Data Representation in using 3D Radar 13

3 System Design **15**
3.1 Background and Related Works 15
3.2 System Description 16

4 Learned Depth Estimation for 3D Imaging Radar **19**
4.1 Introduction 19
4.2 Related Work 21
4.2.1 Radar Imaging Systems 21
4.2.2 Learned Depth Estimation from Images 23
4.2.3 Mapping with Radar 23
4.3 Method 24
4.3.1 Data Representation 24
4.3.2 System Overview 25
4.3.3 Depth Map Estimation 25
4.3.4 Out-of-range Invalid Points 26
4.3.5 Loss Function 27

5 Experiments and Evaluation 29
5.1 Experiment Setup . . . 29
5.2 Depth Estimation . . . 29
5.3 Out-of-Range Classification . . . 31
5.4 Failure Cases . . . 31

6 Downstream Robotics Tasks using Learned Frontend 35
6.1 Occupancy Mapping . . . 35
6.2 Body Frame Velocity Estimation . . . 36
6.3 SLAM through Structured Surfaces . . . 37
  6.3.1 3D Plane Parameterization . . . 38
  6.3.2 SLAM System Setup . . . 40
  6.3.3 Qualitative Results . . . 41

7 Conclusion 43
7.1 Contribution . . . 43
7.2 Discussions and Future Work . . . 43

List of Figures

1.1 (a) Difficult environment for laser and visual systems, were laser are affected by particles in the smoke, and visual feature matching fails due to haze conditions. (b) Well-explored automotive spinning FMCW radar. (c) Phased-Array Imaging radar. 1

1.2 Proposed System for Multi-model sensor platform. 2

2.1 Toy Example for a factor graph in the context of SLAM from [10]. 6

2.2 Illustration of Range Detection for FMCW Radar 9

2.3 Illustration for Angle Estimation 10

2.4 Example Angle FFT result. Two black circles denotes two objects at ±10° individually. 11

2.5 Different methods of increasing RX measurements. Gray circles are TX antennas. White circles are RX antennas. 12

2.6 Example Antenna pattern of **(a)** a short range automotive radar, and **(b)** a long range automotive radar. Image taken from [34]. 12

2.7 Example 2D Heatmap for range and angle. Figure taken from [31]. 13

3.1 Illustration of system time synchronization. A solid arrow notates that the source is querying the target for time. A dashed arrow notates that the source is triggering the target for data collection. 17

3.2 The Sensor Rig we will use to collect the dataset. 17

4.1 Illustration of noise in elevation and azimuth axis. Connected circles denote correspondence between figures. (a) Radar intensity volume backprojected to Euclidean space (plasma colormap), and corresponding LiDAR scans in blue. Radar measurements only corresponding to the center elevation slice in the intensity volume are visualized. (b) Radar intensity volume in native spherical coordinate system. Consistent measurement in elevation axis shows the inability to resolve elevation angle with clarity. Elongated region along azimuth axis shows the difficulty in resolving accurate azimuth angle. (c) Reference LiDAR depth map. (d) Radar depth map obtained from the depth reading at highest radar returns along the range axis on the original intensity volume. 20

4.2 An example of reference LiDAR depth map (top row) and inferred radar depth map (second row), with their view point marked on the map (black bounding box). Occupancy mapping using inferred radar depth map (yellow), compared with that of LiDAR depth map (blue) demonstrates overall ability to generalize in the indoor scenario. Ceiling and floor removed for visual clarity. 22

4.3 3D convolutional network architecture for depth regression and out-of-range classification. The network has a U-net structure for the primary section of the network. The classification has kernel and stride sizes designed to reduce the range dimension to 1. 24

5.1 Qualitative performance on indoor scenarios. The first row shows LiDAR depth maps, the second row shows inferred radar depth maps. Brighter is farther, $r_{\max} \approx 7.58m$. Note that out-of-range pixels are not masked for visual clarity. The third row contains unprojected points and raw radar intensity volume. In the unprojected points figure, LiDAR points are colored blue, radar points are colored yellow. The original density of LiDAR points are used for better visualization of the scenario. In the radar intensity volume, brighter indicates higher intensity. Bounding boxes indicates correspondence between images: green bounding boxes show estimations that are visible in the raw radar volume; black bounding boxes indicate environment features that are difficult to perceive in the raw radar volume. The figure shows that our method is able to generate visually accurate results, and capture floors and ceilings that are barely visible in the original intensity volume. 32

5.2 Examples where the system performed poorly. From top to bottom: LiDAR depth map, radar depth map, and depth map unprojected into 3D space. In the unprojected points figure, LiDAR points are painted blue and unprojected radar points are painted yellow. Connected boxes show correspondence. The red and black bounding box shows situations where our method fails. The yellow structure inside the black bounding box in LiDAR depth map is from dangling wires from the sensor rig. Looking from other view points, the occluded part is a door. 33

6.1 From top to bottom: Mapping using LiDAR depth map limited to radar FoV; Mapping using inferred depth estimation of radar; Mapping using CFAR but only the first peak along each beam of direction. All of the maps have ceiling removed naïvely by thresholding over $0.8m$; the floors exist at around $-0.8m$. For the radar maps, darker denotes lower z value. Therefore brighter color usually denotes walls, and large portion of dark region indicates floors. Mapping using CFAR results in noisier maps due to the sparse nature of the detector and the false positive detections that appear independent of adjacent structures. 36

6.2 Illustration of Factor Graph used in the proposed system. x and π represent robot poses and planes. Square factors are the odometry factors, and circular factors are the plane observation factors. Black plane observation factors are direct co-planar observation, red are orthogonal factors, and blue are parallel factors. Note that due to the relative formulation, subsequent plane observations result in ternary factors that also connect to the first poses that the planes were observed. 41
6.3 System Setup Flowchat. 41
6.4 Left: Mapping results. Different planes are shown with different colors. Right: Odometry results. Achieved visible drift reduction when compared to odometry (blue). 42

List of Tables

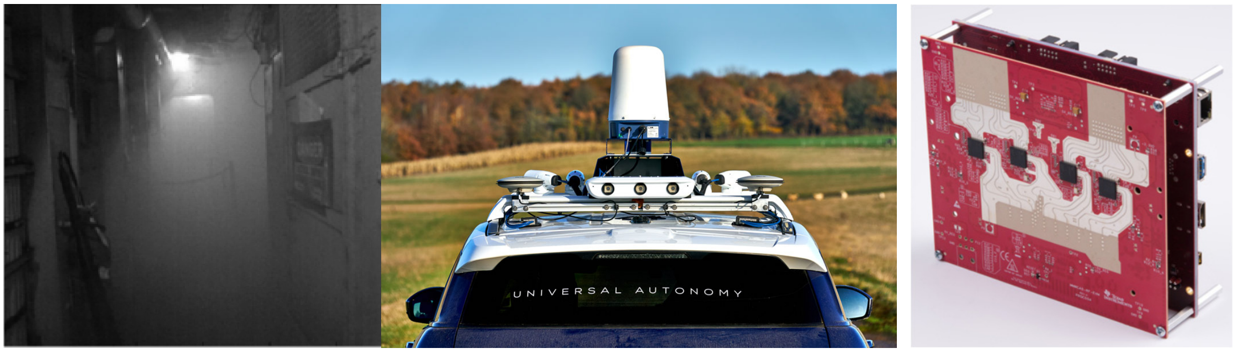
5.1 Depth Evaluation Metrics 30
5.2 Quantitative Results on Depth Estimation 30
5.3 Quantitative Results on Pixel Classification 31

6.1 RMSE for Body Frame Velocity Estimation 38

Chapter 1

Introduction

1.1 Motivation



(a) (b) (c)

Figure 1.1: (a) Difficult environment for laser and visual systems, were laser are affected by particles in the smoke, and visual feature matching fails due to haze conditions. (b) Well-explored automotive spinning FMCW radar. (c) Phased-Array Imaging radar.

One of the reasons for recent successes in simultaneous localization and mapping (SLAM) systems is a thorough understanding of the sensors used for sensor fusion. Consequently, modern state estimation systems rely on a *standard suite* of sensors that includes visual, inertial, and laser-based sensors, sparing a few exceptions such as the Doppler velocity log (DVL) and thermal cameras in specific applications such as underwater and subterranean navigation [23, 60]. Unlike vision and LiDAR sensors that are impacted by low lighting and smoke conditions, the radar system is robust in both extreme weather conditions in the outdoor and visually degraded indoor environments, while uniquely providing relative velocity information [15].

There exists a plethora of work [1, 7, 54] that explores the spinning frequency modulated

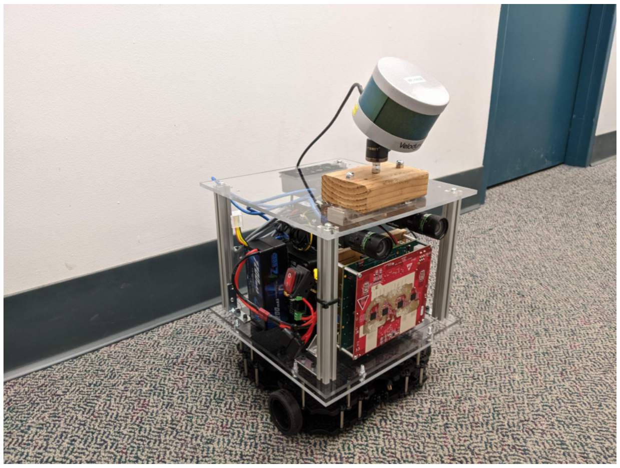


Figure 1.2: Proposed System for Multi-model sensor platform.

continuous wave (FMCW) radar, pioneered by the Oxford RoboCar dataset [38] with much success in automotive applications. In contrast, there are still relatively few work that considers the potential of the short range imaging radar. Compared to spinning FMCW radars used in the automotive settings, short range system-on-chip imaging radars have features that are appealing especially for indoor environments. They provide 3D information, are lightweight and less expensive, have a smaller form factor, and require less power to operate. This makes them suitable for vehicles with limited carrying and power capacity that are commonly seen in indoor scenarios [31].

While the advantages of Imaging radar are prevalent, imaging-radar-based perception is still a challenging and unsolved problem. Radar measurements are notoriously noisy. As opposed to the spinning automotive radars, which realistically offer 1D radar measurements in range, and provide accurate angular information through encoders from its rotating components, 3D imaging radar gives measurement in azimuth, elevation and range domain through static antenna placement and signal processing. The extra dimensions offers more possibilities but also introduces complications and complexities in understanding radar measurements.

These sensor limitations and benefits motivate our research on identifying viable methods to meaningfully use imaging radar measurements

1.2 Scope and Approach

In this thesis, we propose a learned frontend method for localization and mapping using 3D imaging radar sensors. We based these methods on a low cost, off-the-shelf short range imaging radar.

1. The first part of thesis discusses the design of a multi-model sensor platform, focused on localization and mapping. This system is shown in Fig 1.2.
2. The second part of this thesis explores a novel approach for estimating depth from raw radar measurements through LiDAR supervision. We show that the proposed method is able to handle noisy background noises in radar measurement, as well as angle ambiguity induced from antenna placement.
3. The third part of this thesis applies the proposed frontend methods to downstream robotics tasks. Through doppler velocity estimation, probabilistic occupancy mapping, and simultaneous localization and mapping through planes and structural constraints, we demonstrate the effectiveness of the proposed method.

1.3 Organization and Contributions

This thesis is organized as follows. In Chapter 2, we cover the background theory on SLAM, factor graphs, and millimeter wave (mmWave) radar. In Chapter 3, we describe the design of the multi-modal sensor platform. In Chapter 4, we describe the theory and implementation of our learning-based approach to radar perception. Chapter 5 contains the experiment results for the proposed method. Chapter 6 describes the various downstream robotics tasks we have applied the frontend to. In Chapter 7, we conclude this thesis, and propose future work to further enable radar based perception for robot navigation.

Our main contribution outlined in this thesis are as follows:

1. An experimental sensor rig for radar perception, and upcoming dataset for evaluating imaging radar perception in indoor environments.
2. A novel depth estimation method for radar.

Chapter 2

Background

In this chapter, we introduce the topics of simultaneous localization and mapping (SLAM), and millimeter wave (mmWave) frequency modulated continuous wave (FMCW) radar. We begin by providing an overview of the traditional state estimation problem, then we introduce factor graphs for solving probabilistic inference problems. We then describe the primary operating principle of mmWave FMCW radar, and then we will briefly introduce the specific subtype of radar used for this thesis.

2.1 Introduction to SLAM

2.1.1 SLAM and Factor Graphs

The goal of the SLAM problem is to infer where the robot is in the world (localization), and construct a map of the environment (mapping). As measurement noise is unavoidable, we cannot hope to recover the true state of the world. However, it is possible to find the state of the world that agrees with the measurements and robot's motion model the most. There are primarily two frameworks for state estimation: filtering based approaches such as the Kalman filter family [27, 46, 50], and smoothing [49].

Smoothing, otherwise known as optimization-based SLAM, solves a least-squares problem. This formulation preserves the history of measurements, and in the presence of linearization errors, are often more accurate than filtering methods. Recent methods for smoothing and mapping are first presented in SquareRoot SAM [9]. Subsequent work by Kaess et al. [25, 26] developed a strategy that can efficiently incorporate new measurements into the solver without the need to recalculate the entire system. For a high level summary, Cadena et al. provides a comprehensive overview of recent SLAM progress in

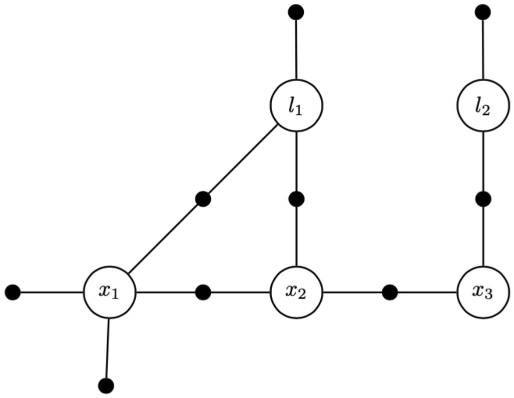


Figure 2.1: Toy Example for a factor graph in the context of SLAM from [10].

[4].

Given the set of robot states $\mathcal{X} = \{\mathbf{x}_m\}, m \in \{1, \ldots, M\}$, landmark locations $\mathcal{L} = \{\mathbf{l}_n\}, n \in \{1, \ldots, N\}$ and a set of measurements $\mathcal{Z} = \{\mathbf{z}_n\}, n \in \{1, \ldots, N\}$, we represent the problem as a factor graph. A factor graph is a bipartite graph comprised of variables to be optimized and factors that constrain the system. In the context of SLAM, the variable nodes represent the states we wish to estimate and the factors are the measurements from the sensors. A toy example from [10] is shown in Fig 2.1.

We wish to compute the *maximum a posteriori* (MAP) estimate of the above problem, which predicts variable values that maximally agree with the given measurements:

$$\begin{aligned}\tilde{\mathcal{X}}, \tilde{\mathcal{L}} &= \underset{\mathcal{X},\mathcal{L}}{\arg\max}\, p(\mathcal{X}, \mathcal{L} \mid \mathcal{Z}) \\ &\propto \underset{\mathcal{X},\mathcal{L}}{\arg\max}\, p(\mathcal{Z} \mid \mathcal{X}, \mathcal{L}) p(\mathcal{X}, \mathcal{L}) = p(\mathcal{X}, \mathcal{L}, \mathcal{Z}).\end{aligned} \tag{2.1}$$

We solve this problem by minimizing the negative log likelihood form of equation (2.1),

$$\log p(\mathcal{X}, \mathcal{L}, \mathcal{Z}) = \underbrace{\log p(\mathbf{x}_0)}_{\text{prior}} + \sum_{n=1}^{N} \underbrace{p(\mathbf{z}_n \mid \mathbf{x}_{a_n}, \mathbf{l}_{b_n})}_{\text{landmark observation}} + \sum_{t=1}^{M-1} \underbrace{p(\mathbf{z} \mid \mathbf{x}_{t-1}, \mathbf{x}_t)}_{\text{odometry}}, \tag{2.2}$$

where data association $\mathcal{D} \in \mathbb{R}^{N \times 2}$ between measurements, robot poses and landmarks is known. It is common practice to make a simplifying assumption that measurements are drawn from a zero-mean Gaussian distribution, so we rewrite all factors now in the following form:

$$\phi_i(X_i) \propto \exp\left(-\frac{1}{2}||h_i(X_i) - \mathbf{z}_i||^2_{\Sigma_i}\right), \tag{2.3}$$

where $\phi_i(\cdot)$ represents a single factor, $X_i \in X = \{\mathcal{X}, \mathcal{L}\}$ represents all the relevant variables associated with this factor, and $h_i(\cdot)$ the measurement prediction function. With this Gaussian assumption, the objective can be written as follows:

$$\tilde{X} = \arg\min_{X} \sum_{n=i}^{N} ||\mathbf{z}_i - h_i(X_i)||^2_{\Sigma_i}. \tag{2.4}$$

By linearizing the measurement function using first order Taylor expansion, we can separate the function into the value evaluated at X_i^0, Jacobian H of measurement function with respect to the state vector, and the state update vector Δ_i:

$$h_i(X_i) = h_i(X_i^0) + H_i\Delta_i, H_i \triangleq \left.\frac{\partial h_i(X_i)}{\partial X_i}\right|_{X_i^0}. \tag{2.5}$$

the objective function can further be rewritten into an iterative nonlinear least squares problem where we solve for the state update vector at each iteration:

$$\Delta^* = \arg\min_{\Delta} \sum_{i} ||\Sigma_i^{1/2} H_i \Delta - \Sigma_i^{1/2} \left(z_i - h_i(X_i^0)\right) ||_2^2, \tag{2.6}$$

This nonlinear least squares optimization can be solved directly, but requires a good initial estimate. A variety of algorithms are available for solving this problem, including steepest descent, Gauss-Newton, Levenberg-Marquardt, and Dogleg Minimization [10]. Additionally, aforementioned work by Kaess et al. [26] incorporated incremental update techniques to the matrix solver, which can be used for real-time operation in the SLAM context.

2.1.2 Frontend of SLAM

The previous section introduced the graphical backend of a SLAM system. Such system often assumes that the graph is known *a priori*, namely measurement functions, consistent measurements that are drawn from Gaussian distributions, as well as data association between state variables. The part of the system that prepares such information is often referred to as the frontend of the SLAM system.

Different sensors would have different frontends. Visual SLAM systems typically relies on feature descriptors to match between different observed visual keypoints in the environment [41], Laser-based SLAM systems uses a hybrid of scan alignment and feature matching to obtain between frame motion [57], wheel encoder uses the motion model of the vehicle to obtain wheel odometry, and etc.

2.2 mmWave FMCW Radar

2.2.1 Introduction to mmWave FMCW Radar

Radar systems transmit electromagnetic (EM) waves signals that are reflected by the objects in the path, and by capturing the reflected signal, a radar system can determine the objects' bearing, range, and velocity relative to the radar. Millimeter wave (mmWave) radar is a special class of radar that uses short-wavelength EM waves [22]. mmWave enables the miniaturization of antenna, and therefore the possibility of indoor robot navigation through radar.

In this section, we introduce the underlying principle for mmWave FMCW radar to measure range, velocity, and angle. We will then briefly introduce the specific type of imaging radar that this thesis focuses on: **cascaded imaging radar in MIMO short range mode**. Compared to conventional radar sensors, this type of radar offers significantly better angular resolution and opens up new possibilities of radar perception.

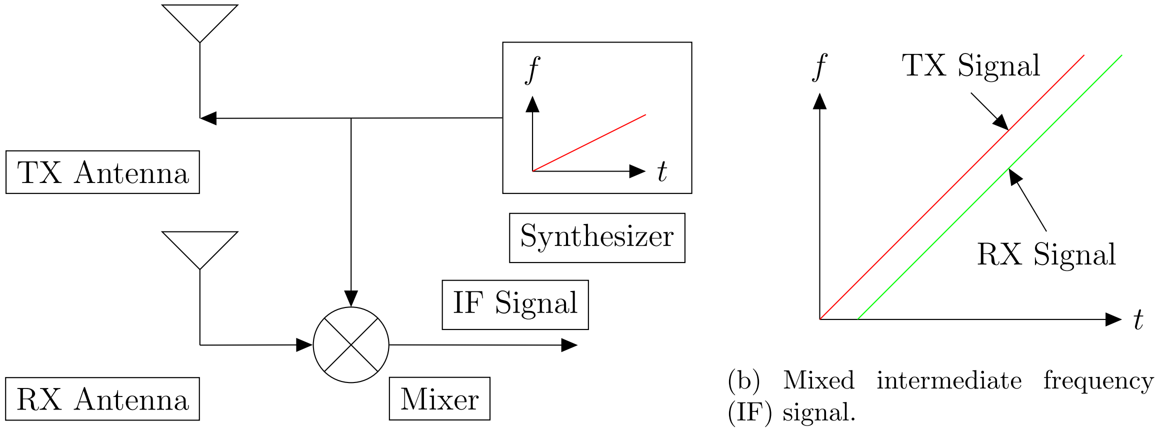
Range Detection

Frequency-Modulated Continuous Wave (FMCW) radar transmits frequency modulated EM wave continuously. FMCW radar has the advantage of being able to determine target range compared to non frequency-modulated radar. Consider the very simple case of a single transmit antenna, a single receive antenna, and a single static object. This radar system is illustrated in Fig 2.2a, the object is omitted from the illustration. In a single detection cycle, the synthesizer generates a chirp and is transmitted by the transmit (TX) antenna. The reflection of the chirp is captured by the receive (RX) antenna. The mixer combines the TX and RX signals to produce an intermediate frequency (IF) signal in Fig 2.2b. Notice that the RX chirp is a time-delay version of the TX chirp, and the time delay can be derived as

$$\Delta t = \frac{2d}{c}, \tag{2.7}$$

where d is the distance to the detected object and c is the speed of light.

Since the horizontal distance between the two chirp is fixed, and that the ramp of the chirp stays the same, the vertical difference between the two chirps is also fixed. This leads to a tone with a constant frequency in the IF signal, which is rather easy to be detected in the frequency domain. If there are more than a single object, there would be multiple



(a) Example FMCW Radar Block Diagram.

(b) Mixed intermediate frequency (IF) signal.

Figure 2.2: Illustration of Range Detection for FMCW Radar

reflected chirps, and multiple constant frequency tones would be observed in the frequency domain.

Velocity Measurement

In order to measure velocity, the FMCW radar transmits two chirps separated by T_c. If we assume the object is moving at a much slower speed v than the speed of light, then each reflected chirp will have peaks in the same location, but with a different phase. The measured phase difference corresponds to the speed of the object relative to the radar.

Consider the phase ϕ_0 at the beginning of TX signal, where λ is the wavelength, and the phase ϕ_1 at the beginning of the RX signal:

$$\phi_0 = 2\pi f_c \Delta t = \frac{4\pi d}{\lambda}, \phi_1 = \frac{4\pi(d + vT_c)}{\lambda} \tag{2.8}$$

$$\Delta\phi = \frac{4\pi v T_c}{\lambda} \Rightarrow v = \frac{\Delta\phi\lambda}{4\pi T_c}. \tag{2.9}$$

Since the velocity measurement is based on phase difference, the measurement is unambiguous only if $|\Delta\phi| < \pi$, and consequently $|v_{\max}| < \lambda/(4T_c)$. When there are multiple moving objects with different speed at the same distance from the radar, it is required to transmit more than two chips and additionally perform a second Doppler-FFT to resolve the two objects.

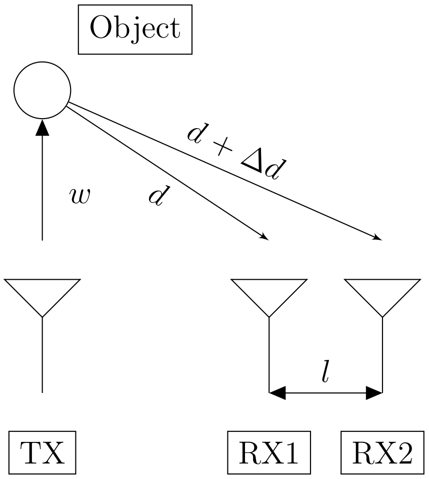


Figure 2.3: Illustration for Angle Estimation

Angle Measurement

Angle estimation is based on the concept that objects off the center axis results in small difference in distances to different RX antennas. This difference is small enough that it results in a phase difference rather than a time-delay IF signal. We consider the example in Fig 2.3 where we need to compute the Angle of Arrival (AoA) of the target object. With the same principle from velocity measurement, we can derive the phased difference to be the following:

$$\phi_0 = 2\pi f_c \Delta t = \frac{2\pi(w+d)}{\lambda}, \phi_1 = \frac{2\pi(w+d+\Delta d)}{\lambda} \Rightarrow \Delta\phi = \frac{2\pi\Delta d}{\lambda}. \tag{2.10}$$

Notice the phase difference is slightly different compared to equation 2.8 since the distance from the TX to the object is the same and therefore omitted. Under the assumption of a planar wavefront geometry so that:

$$\Delta d = l \sin(\theta)$$

where l is the distance between the antennas. The angle of arrival θ can be computed as follows:

$$\theta = \sin^{-1}\left(\frac{\lambda\Delta\phi}{2\pi l}\right). \tag{2.11}$$

In reality, where the computation happens over multiple TX and RX pairs, $\sin(\theta)$ is approximated with θ, and as a result, the estimation accuracy depends on AoA and is more accurate when AoA is small [22].

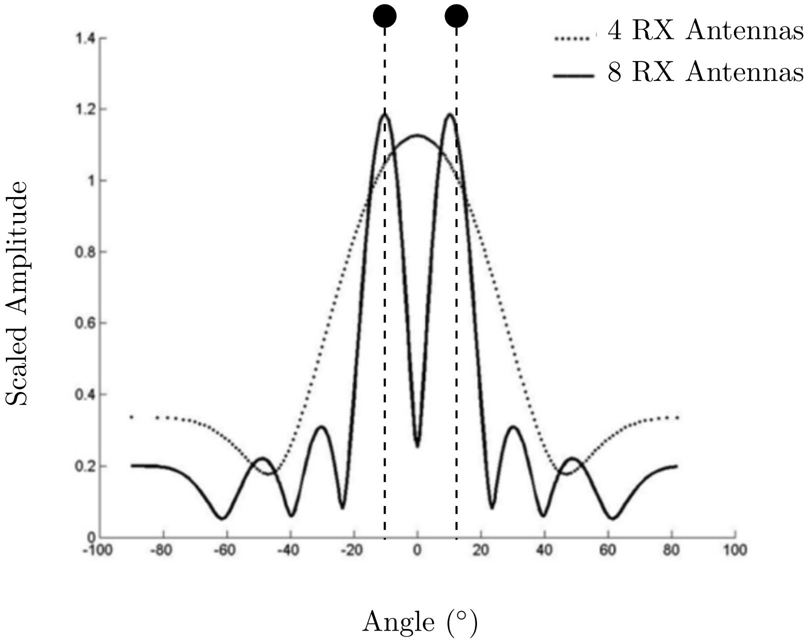


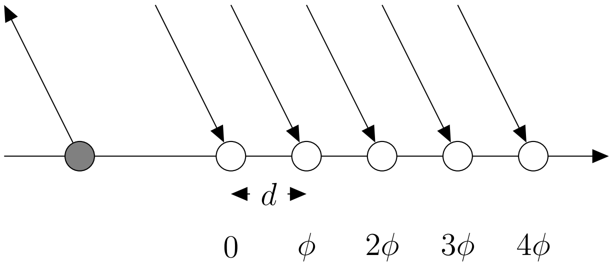
Figure 2.4: Example Angle FFT result. Two black circles denotes two objects at ±10° individually.

2.2.2 Imaging Radar using Cascaded FMCW mmWave Radar Systems in MIMO Configuration

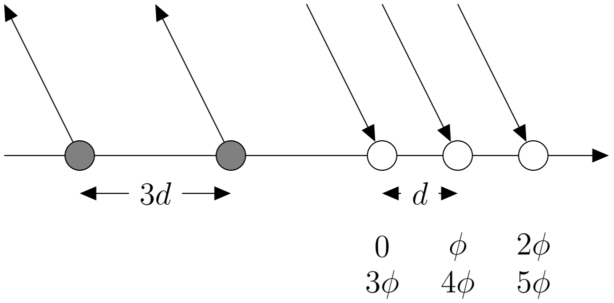
Multiple Input Multiple Output (MIMO) radar system is a novel radar method where each transmit antenna radiates an arbitrary waveform independently of each other. Theoretically, MIMO is capable of improving the angular resolution of mmWave Radars.

In regular radar operations, people rarely enumerate antenna pairs and perform trigonometry individually. Instead, angle-FFT is performed [45], and an example result is illustrated in Fig 2.4. Typically, a higher amplitude is indicative of the presence of an object. In our example illustration here, there are two objects placed at $+10°$ and $-10°$ individually. As shown in the dotted lines, having only 4 RX antennas is unable to resolve the two objects, but having 8 RX antennas would give a better resolving power to the radar and results in a clearer resolution of the two targets.

There are different ways of increasing the effective RX antenna numbers. One would be to simply add more RX antennas, and additional measurements can be obtained by simply sampling the signals across the RX antennas, as illustrated in Fig 2.5a. Alternatively, an additional TX antenna can bring in additional measurements by multiplexing the RX and TX pairs. Radar that operates using this principle is called Multiple-Input Multiple-Output (MIMO) radars. MIMO radar provides a cost-effective way to improve the angle



(a) Naïvely adding more RX antennas.



(b) Add TX antenna (MIMO).

Figure 2.5: Different methods of increasing RX measurements. Gray circles are TX antennas. White circles are RX antennas.

resolution of the radar.

A cascaded sensor is a type of radar that uses multiple radar chips as one, but achieves better performance at the cost of lower frame rate, higher data volume, and higher power consumption [31]. Cascaded Radar devices can support long-range radar (LRR) applications through beam-forming, as well as medium-range (~150m) (MRR) and short-range high-angular resolution sensing (SRR). We show an example comparison of radar antenna patterns between short range and long rang mode in Fig 2.6.

Imaging radar implemented using cascaded radar chips that transmit FMCW mmWave EM wave operating in short range MIMO mode is the point of focus of this thesis. Long

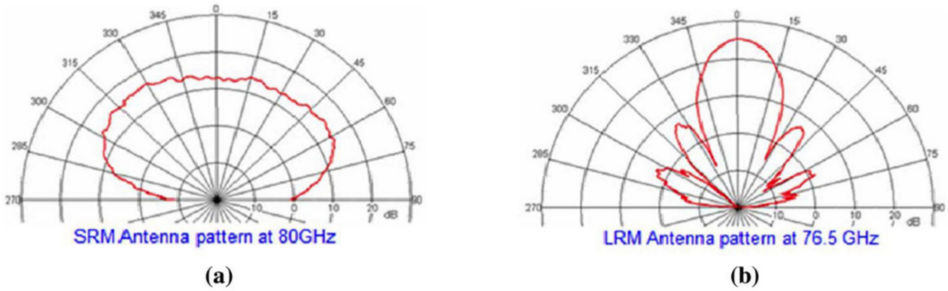


Figure 2.6: Example Antenna pattern of **(a)** a short range automotive radar, and **(b)** a long range automotive radar. Image taken from [34].

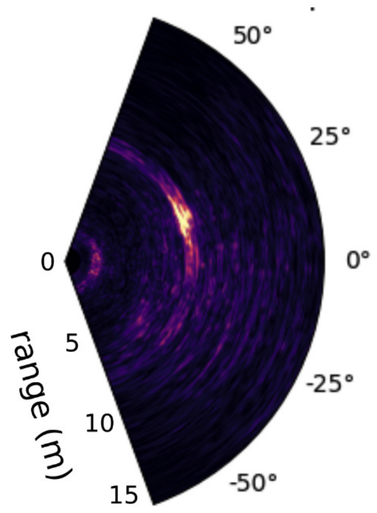


Figure 2.7: Example 2D Heatmap for range and angle. Figure taken from [31].

and medium range sensing give up angular field of view and resolution for much more elongated gain field that reaches out very far. Consequently, the focus of long and medium range sensing are typically object detection and target tracking. As the indoor environment typically does not require a far reaching sensor, the radar operates in short range mode, which trades range for high angular resolution. We believe short range imaging radar has the potential for being a versatile sensor in situations where conventional sensors will fail, and provide far richer and more accurate measurements than sparse detections.

2.2.3 Data Representation in using 3D Radar

The author's generalization of radar data representations for radar perception is as follows:

1. Raw data: raw signals captured by antennas.
2. Volumetric data: semi-processed dense data. The output of Angle and Range-FFT.
3. Sparse data: sparse targets. Results of applying target detectors on the above data.

While raw data and sparse data are easier to be interpreted, as they are simply waveforms and sparse 3D points, volumetric data is rather different from common perception representation seen in robotics. Consider the previous Angle-FFT output example in Fig 2.4. If we incorporate range measurement into this scaled amplitude vs angle plot, the result would be a 2D heatmap in the polar coordinates (r and θ). We show an example of such heatmap in Fig 2.7, where a clear peak is present around $8m, 25^\circ$. If we extend this representation to 3D by incorporating the elevation angle, the representation would be akin to a 3D heat volume.

Raw data are seen more in very short range high resolution imaging tasks such as [52], while sparse data are seen more in robot navigation tasks [37]. Until recently there has not been much work in utilizing the volumetric data from 3D imaging radar. With the advent of higher resolution imaging radar through cascaded sensors, we hope the use of volumetric data can provide denser and more accurate information for robot navigation.

Chapter 3

System Design

In this chapter, we describe the design of a multi-modal sensor platform for collecting 3D Imaging radar data. In Section 3.1, we introduce the background and the current state of publicly available radar perception datasets. In Section 3.2, we describe our sensor platform, including the sensor selection and system setup.

3.1 Background and Related Works

With the advancement of autonomous driving in the automotive industry, there has been significant research effort going into mechanically spinning FMCW radar. There exist at least multiple publicly available robotics dataset containing this type of radar, including but not limited to [3, 5, 29, 53]. These datasets feature urban driving scenarios only, and the mechanically spinning radar only provides dense 2D scans in the xy plane only, making the dataset unsuitable for developing methods in the 3D environment.

High resolution imaging radar has been gaining traction as well: TJ4DRadSet, and K-Radar are two recent works that incorporates 4D imaging radar that provides range, azimuth, elevation, and speed on an automotive platform [43, 61]. However both datasets put heavy emphasis on object detection rather than precise perception through radar. TJ4DRadSet only provides sparse radar point detections, limiting researchers to the degree of signal processing and target detection algorithms implemented onboard the sensors.

The ColoRadar dataset [31] is the closest work to the dataset we are aiming at. ColoRadar dataset provides LiDAR, inertial, one cascaded, and one single chip radar on a handheld sensor rig performing drone-like motion. The dataset provided multiple indoor and outdoor runs, while making different levels of radar data available: raw ADC values, heatmap, and sparse point targets from onboard target detection algorithms. One limitation to this dataset is the scarcity of sensor modalities. Having access to only LiDAR point clouds are sometimes confusing and not enough to understand the corresponding radar measurement. The dataset also lacks a source of motion estimate that are robust to challenging visual laser environment. Since imaging radar-based SLAM algorithms are still in their infancy, not having a source of good motion estimate that are robust to smoke and visually challenging environment hinders the ability to test primitive imaging radar-assisted SLAM methods in a realistic setting.

3.2 System Description

Compared to ColoRadar, we aim at providing two additional functionalities: visual representation of the scene through a stereo machine camera, and wheel odometry. Our sensor platform is shown in Fig 3.2. We build the sensor rig on top of a Turtlebot 3 Carrier Variant[1] to provide wheel odometry. The sensor rig is detachable from the Turtlebot, and independently operable as a handheld rig. The sensor rig contains the following sensors:

- Velodyne VLP-16 LiDAR,
- Two Point Grey BFLY-PGE-50S5C-C Machine Camera,
- Epson G364 IMU,
- Cascaded Imaging Radar Sensor: Texas Instrument MMWCAS-RF-EVM.

The sensor rig is equipped with an Intel i7 NUC[2] for data logging. We control and synchronize the IMU, Velodyne, and the NUC by generating a simulated GPS signal on a Teensy 4.0 [3]. Both cameras and the radar is time-synchronized with the NUC over network, and we trigger the cameras using pulses from the Teensy. The time synchronization is illustrated in Fig 3.1. Note that the IMU does not necessarily obtain true time from the Teensy. The Teensy sends a Pulse Per Second (PPS) signal to the IMU to reset the onboard counter and mark the beginning of a new second. When IMU transmit the recorded package to NUC, the IMU driver on the NUC converts the IMU counter value into millisecond elapsed from the beginning of the second.

We currently plan to collect indoor data in Newell Simon Hall both on the Turtlebot and off the ground vehicle performing 6 DoF motion. We plan to release the dataset to the public. Note that the remaining work of this thesis are build on top of the ColoRadar dataset and not the data collected from this platform.

---

[1] https://emanual.robotis.com/docs/en/platform/turtlebot3/locomotion/

[2] https://www.intel.com/content/www/us/en/products/details/nuc.html

[3] https://www.pjrc.com/store/teensy40.html

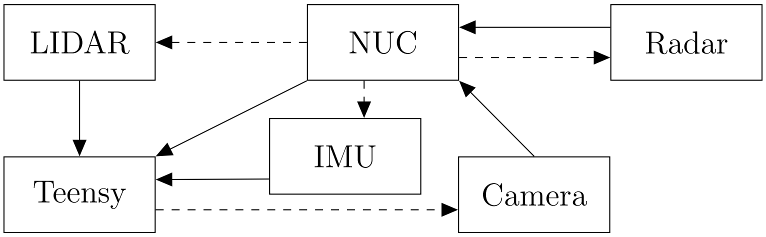

LIDAR
NUC
Radar
IMU
Teensy
Camera

Figure 3.1: Illustration of system time synchronization. A solid arrow notates that the source is querying the target for time. A dashed arrow notates that the source is triggering the target for data collection.

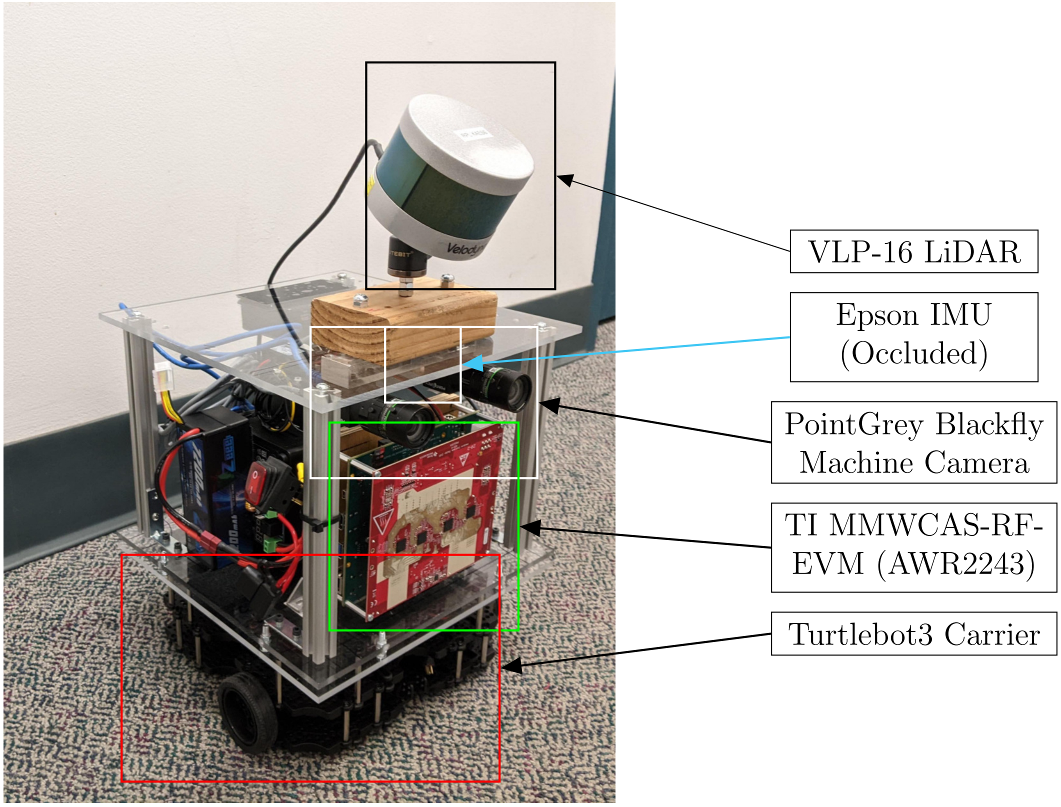

VLP-16 LiDAR
Epson IMU (Occluded)
PointGrey Blackfly Machine Camera
TI MMWCAS-RF-EVM (AWR2243)
Turtlebot3 Carrier

Figure 3.2: The Sensor Rig we will use to collect the dataset.

Chapter 4

Learned Depth Estimation for 3D Imaging Radar

Despite the intricate engineering design that went into imaging radars to make them better and more accurate than their predecessors, accurate and moreover *dense* perception through radar is still a very challenging task due to the noisy measurements, and uncertainties induced by radar-specific designs. In this chapter, we propose a novel method for imaging radar perception through LiDAR supervision. In Section 4.1 we will provide an overview of the challenges in radar perception, as well as the specific contribution of our work. In Section 4.2, we will provide a literature review for existing radar sensing and mapping methods. I will then delineate the proposed methods in Section 4.3.

4.1 Introduction

Imaging radar are robust to smoke and dust environments where visual and laser based system fail. Compared to single chip radars, cascaded imaging radar provides dense measurement that are desirable for mapping out environments. Imaging radar however, has distinct characteristics that make it difficult to work with. Beyond the noisy measurements typically observed in spinning radars, since the angular information of detected targets is resolved through antenna arrays, the antenna placement affects the resolution and accuracy of the angular dimensions asymmetrically. Fig. 4.1 is an example comparison between the radar heatmap obtained post analog to digital converter (ADC) and the LiDAR measurement. It illustrates that the radar is not only unable to resolve azimuth direction with clarity, but also provides no clear association along the elevation axis: all range-azimuth slices at different elevations are nearly indistinguishable.

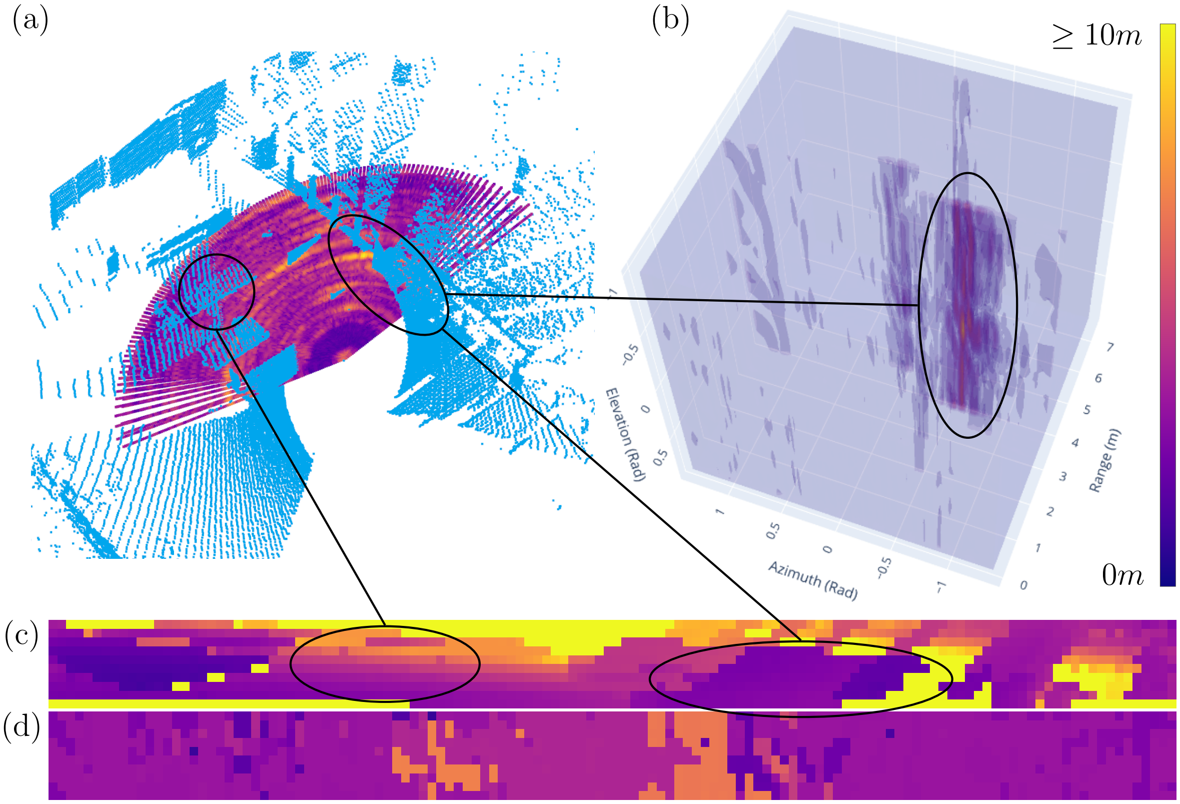


Figure 4.1: Illustration of noise in elevation and azimuth axis. Connected circles denote correspondence between figures. (a) Radar intensity volume backprojected to Euclidean space (plasma colormap), and corresponding LiDAR scans in blue. Radar measurements only corresponding to the center elevation slice in the intensity volume are visualized. (b) Radar intensity volume in native spherical coordinate system. Consistent measurement in elevation axis shows the inability to resolve elevation angle with clarity. Elongated region along azimuth axis shows the difficulty in resolving accurate azimuth angle. (c) Reference LiDAR depth map. (d) Radar depth map obtained from the depth reading at highest radar returns along the range axis on the original intensity volume.

Classically, the phase following the ADC data processing is a target detection approach through a detector such as the constant false alarm rate (CFAR) [14] or its variants. These methods filter the heatmap into a sparse set of point targets by detecting peaks based on an estimate of local noise. While this reduces the dimensionality of the observation drastically, a substantial amount of potentially valuable information is thrown out. Target detection also requires expert tuning of parameters to identify targets correctly. In contrast, we wish to utilize information in the radar intensity volume to produce a *dense* estimate that addresses the aforementioned issues.

In particular, we present a learning-based method that regresses a dense depth map from the intensity volume of 4D imaging radar data. Our main contributions are two-fold:

1. A learned method for inferring a depth map for single frame imaging radar in a generalized indoor scene using LiDAR supervision.
2. We demonstrate the effectiveness of our method through the downstream tasks of 3D occupancy mapping, body frame velocity estimation, and SLAM through planes extracted from our learned depth estimation and wheel odometry. An example occupancy mapping result is shown in Fig. 4.2.

Additionally, our method shows the potential for presenting 3D imaging radar in the form of more popular sensors in robotics. Following a literature review in Section 4.2, we present and discuss our method in Section 4.3.

4.2 Related Work

4.2.1 Radar Imaging Systems

There exist several well-established short-range mmWave imaging radar systems and datasets [17, 39, 52]. They demonstrate high resolution radar imaging, however, they only operate over very short ranges, or require a bulky sensor setup and radar absorbing materials in the background, making them impractical for navigation tasks. Another approach is to utilize motion to simulate a synthetic aperture radar [44, 47, 51]. While the results are promising, especially in the case of *MilliPoint* where accurate elevation information is also available as an output, one shortcoming of these methods is that the resulting imaging depends on multiple observations, which poses difficulty for tasks when accurate motion information is not readily available.

Recently, works have leveraged deep learning in the context of mmWave radar data. One line of work addresses radar-camera fusion using automotive spinning FMCW radars

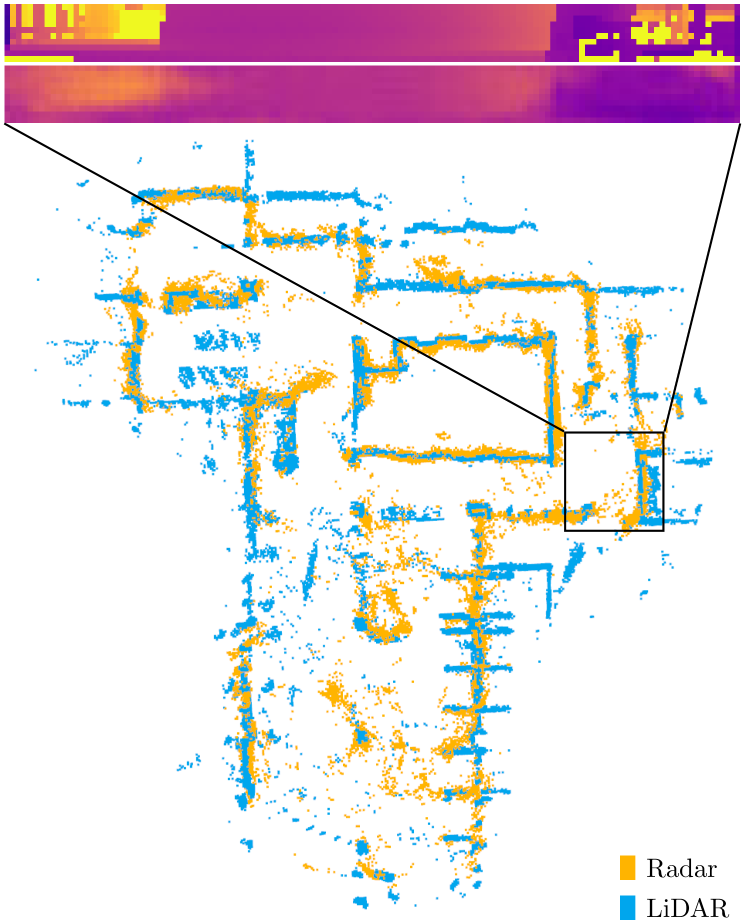


Figure 4.2: An example of reference LiDAR depth map (top row) and inferred radar depth map (second row), with their view point marked on the map (black bounding box). Occupancy mapping using inferred radar depth map (yellow), compared with that of LiDAR depth map (blue) demonstrates overall ability to generalize in the indoor scenario. Ceiling and floor removed for visual clarity.

[16, 35, 36], while others show reasonable success in learning human poses from radio frequency (RF) signals [58, 59] or filtering information from classical detectors [33]. Most notable is the application of a conditional generative adversarial network (cGAN) for imaging radar depth estimation proposed in [18, 48]. These methods achieve impressive results for depth map inference from radar intensity volumes, however both methods are specialized for single class single object estimation, and the method scales poorly when multiple instances of the trained object are seen in the input image. Their ability to generalize to more generic scenes is questionable.

4.2.2 Learned Depth Estimation from Images

Learning-based methods are capable of fitting the function between camera images and their corresponding depth maps [6, 32]. In terms of formulation, our work is closely related to depth refinement since the radar intensity volume can be considered as a coarse estimate of depth. However, we find that depth refinement is typically a subtask in monocular or multi-view stereo depth estimation [55, 56], where the objective is to remove and smoothen artifacts around contours of similar depths. Such refinement processes have access to RGB images that innately encode high-frequency features such as boundaries and edges, providing excellent local information for regression and smoothing of pixel depth, unlike radar. Additionally, due to the noisy nature of resolving angle of arrival from targets in the imaging radar, it is common for multiple peaks to reside along the same single beam of direction due to the noise of multiple adjacent targets. It should be noted that these are not multi-path reflections. Such a characteristic sensor model complicates leveraging structure, such as the association of features or recognition of structures.

It is also well-known that radar has very different penetrative and reflective characteristic from laser and camera sensors [15]. Therefore, ground truth generation for radar is not a straightforward process for learning, and is either difficult, reliant on synthetic data, or assumes that laser measurements are the ground truth.

4.2.3 Mapping with Radar

Limited range and angular resolution, multi-path reflections, and sparse measurements present challenges for 3D mapping using radar. One approach uses handcrafted prior knowledge of environment geometry, such as corridor width, to filter radar measurements [11]. With accurate odometry, and good knowledge of the environment, occupancy grids can be built from sparse targets, however these assumptions are rarely true when operating in a visually degraded environment.

Another approach to tackle mapping is to learn a deep model that generates a dense map given a set of radar detections. *MilliMap* [37] stitches multiple frames of scans from a single-chip mmWave radar together in the form of 2D occupancy grids, and uses a conditional GAN to complete the map. While it shows promising results, collecting the large amount of training data required is very costly.

Recent work has shown significant progress and potential in using mmWave imaging radar for robot perception, yet the challenge of extracting dense 3D information from a single frame of radar data has not been sufficiently tackled. This motivates our work to

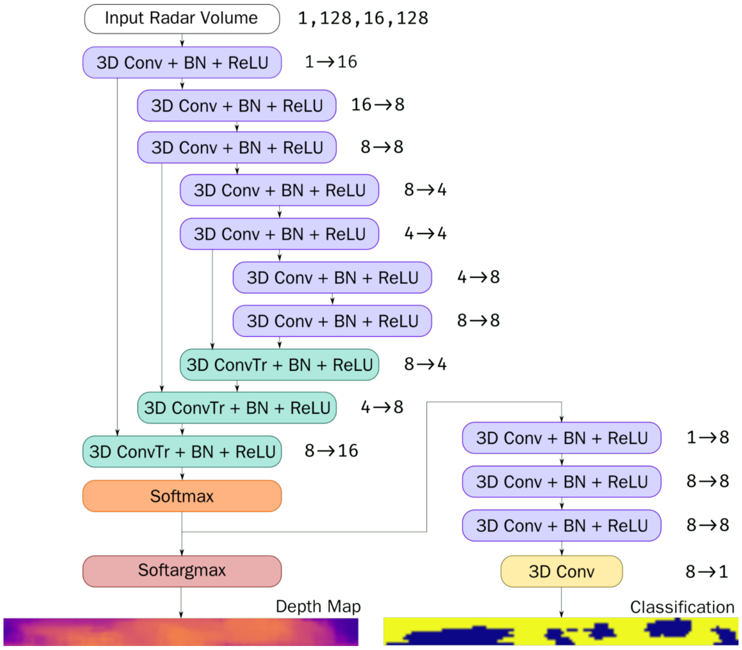

Input Radar Volume
1,128,16,128
3D Conv + BN + ReLU
1→16
3D Conv + BN + ReLU
16→8
3D Conv + BN + ReLU
8→8
3D Conv + BN + ReLU
8→4
3D Conv + BN + ReLU
4→4
3D Conv + BN + ReLU
4→8
3D Conv + BN + ReLU
8→8
3D ConvTr + BN + ReLU
8→4
3D ConvTr + BN + ReLU
4→8
3D ConvTr + BN + ReLU
8→16
Softmax
Softargmax
Depth Map
3D Conv + BN + ReLU
1→8
3D Conv + BN + ReLU
8→8
3D Conv + BN + ReLU
8→8
3D Conv
8→1
Classification

Figure 4.3: 3D convolutional network architecture for depth regression and out-of-range classification. The network has a U-net structure for the primary section of the network. The classification has kernel and stride sizes designed to reduce the range dimension to 1.

create a system that allows for single frame 3D perception for imaging radar, enabling application of imaging radar for robot navigation tasks.

4.3 Method

We propose a 3D convolution based supervised regression model for the task of depth estimation. This section describes the detailed architecture of the proposed network and the data representation used for input and supervision.

4.3.1 Data Representation

The measurements from mmWave radar has four dimensions: range, azimuth, elevation, and with the last dimension consists of intensity and velocity. From this 4D heat-volume, we use the 3D per-voxel intensity as our input. Since radar measures environment by range and angle, this representation is natively in the 3D spherical coordinate system of (r, ϕ, θ) (range, azimuth, elevation). We assume that laser measurements are a good proxy for the true radar measurements, even though mmWave radar has different penetrative characteristics than laser sensors.

There are several ways to formulate radar-laser supervision. One formulation would be to backproject radar points into their Euclidean coordinate representation and then classify individual points. [8]. However due to the spherical coordinate system, adjacent points in the intensity volume grow farther apart with increasing range, which makes it difficult to capture local relationships which should be invariant to the range value. This is made worse with the spare resolution. Second, it is also possible to construct an occupancy volume in the same spherical coordinate system from LiDAR measurements. Adopting a depth map formulation retains local relationships, and is a simpler representation of the sensor model. Compared to the volumetric or unprojected points, the depth map representation loses the ability to distinguish between multiple detections along the same angular orientation, that could be caused by multi-path or behind-the-wall detections. However in the context of indoor mapping, with the detection range of medium range imaging radar at multiple-input multiple-output (MIMO) mode at less than $8m$, this loss of representation does not lose significant information.

The ground truth depth maps are generated by projecting LiDAR scans into a cylindrical depth map. To avoid choppy images caused by calibration issues, we utilize the formulation introduced in [12]. The radar intensity volume is further cropped to have a similar vertical field of view (FoV) as LiDAR.

4.3.2 System Overview

The system comprises primarily of two sections, a depth map regression model and an out-of-range pixel classification model. This pixel classification model accounts for two shortcomings in the radar based regression: 1) a significant number of pixels in the depth map are often out of the radar detection range, due to the relatively short range of imaging radar and 2) large noise in the angular coordinates requires the use of local information (convolution) to discriminate a false return due to targets in adjacent region from a true radar return. The network architecture is illustrated in Fig. 4.3.

4.3.3 Depth Map Estimation

The depth map regression takes heavy intuition from cell-averaging CFAR, which could be viewed as a rectangular filter. However, the parameter tuning for CFAR is time-consuming and hard to evaluate due to the sparse spatial resolution. We recognize the similarity between CFAR and convolution, and the success in learning-based reconstruction, especially in [56] where the feature points are collected into a cost volume trained to generate a probability volume for the estimated depths. We adopt a probabilistic view where the raw intensity volume observed are noisy measurements from which, an underlying true depth can be estimated.

With that intuition, we use a multi-scale 3D convolutional backend for depth regression. We use skip connections to link earlier convolutional blocks to the deconvolution layers, batch normalization and ReLU activations for the deconvolution layers. The last deconvolution layer outputs a 1-channel 3D volume.

We define range, azimuth, and elevation angle bin $\mathbf{r}, \boldsymbol{\theta}, \boldsymbol{\phi}$ of the radar measurement volume and note the original intensity volume as $\mathbf{V}_I$ and output volume at the last deconvolution layer $\mathbf{P}$, $\{\mathbf{V}_I, \mathbf{P}\} \in \mathbb{R}^{|\mathbf{r}|\times|\boldsymbol{\theta}|\times|\boldsymbol{\phi}|}$. To preserve differentiability, we compute the 2D image $\mathbf{I}_r \in \mathbb{R}^{|\boldsymbol{\theta}|\times|\boldsymbol{\phi}|}$ through soft argmax that computes the expected value given the distribution

$$\mathbf{I}_r(\theta, \phi) = \sum_{r=0}^{|\mathbf{r}|} \mathbf{r}(r)\mathbf{P}(r, \theta, \phi) \tag{4.1}$$

4.3.4 Out-of-range Invalid Points

The true radar depth map is limited by the range of space covered by $\mathbf{P}$. Therefore there are depth values in $\mathbf{I}_r$ that should be returned as out of bounds. While we do observe a positive correlation between the standard deviation of probability along the depth axis in $\mathbf{P}$ for the out-of-range pixels, the high false positive and true negative rate requires a more intelligent processing method. Other methods such as photometric confidence proposed in [56], which calculates the confidence in prediction based on the surrounding values of the maxima, were also experimented on to no significant success.

The task here is to reduce $\mathbb{R}^{|\mathbf{r}|\times|\boldsymbol{\theta}|\times|\boldsymbol{\phi}|}$ to $\mathbb{R}^{|\boldsymbol{\theta}|\times|\boldsymbol{\phi}|}$ while maintaining invariance to the position of values along the depth dimension. We propose to use consecutive 3D convolutional blocks with strides and kernel sizes that incrementally reduce the dimension along the depth axis while maintaining the dimensions for azimuth and elevation. Each block has eight channels with batch normalization and ReLU as activation function. The last layer outputs an image for pixel-wise classification $\mathbf{I}_c \in \mathbb{R}^{2\times|\boldsymbol{\theta}|\times|\boldsymbol{\phi}|}$.

4.3.5 Loss Function

The loss function considers both the classification of out-of-range pixels and depth regression. We only consider in-range pixels for the depth estimation task.

$$Loss = l_{\mathrm{BCE}}(\mathbf{I}_m, \mathbf{I}_c) + \sum_{p \in \mathbf{I}_l < r_{\max}} \psi\left(\mathbf{I}_l(p) - \mathbf{I}_r(p)\right) \tag{4.2}$$

Here $p \in \mathbf{I}_l < r_{max}$ denotes the set of pixels that are within maximum range of $\mathbf{r}$, $\mathbf{I}_l$ the LiDAR ground truth, $\mathbf{I}_m \in \mathbb{R}^{2\times|\boldsymbol{\theta}|\times|\boldsymbol{\phi}|}$ the ground truth map for in-range and out-of-range pixels. ψ denotes Huber robust cost [21] function to account for situations where radar measurements detect significantly different objects from LiDAR. l_{BCE} is the binary cross entropy loss.

Chapter 5

Experiments and Evaluation

5.1 Experiment Setup

We evaluate our method on the publicly available mmWave imaging radar dataset ColoRadar [31]. The dataset provides data from an IMU, LiDAR, a Texas Instruments (TI) cascaded imaging radar (AWR2243) operating in MIMO mode, and a single chip radar (AWR1843). The dataset contains sequences through both indoor and outdoor environments as well as ground truth trajectory generated from LiDAR-inertial SLAM methods.

Specifically, we train and test in the *ec_hallways* and *arpg_lab* sequences. These two scenarios are representative of an ordinary building as they travel through corridors and large rooms. The sensor rig performs quadrotor-like motion during data recording. We train the network on sequence 2 of *ec_hallways* and test on the rest of the sequences. Sequence 3 of *ec_hallways* is omitted due to a ~$20s$ duration of dropped radar frame in the middle of the run. Training is performed on an 8GB NVIDIA RTX2070S with a batch size of 16 for 150 epochs using Adam optimizer.

5.2 Depth Estimation

We show some sample outputs in Fig. 5.1, where we compare the LiDAR and inferred radar depth map, the unprojected points of both LiDAR and radar depth map, and the original $\mathbf{V}_I$. Our method generates depth maps that are visually consistent with LiDAR ground truth. As a baseline, our method can capture visually significant peaks in the original radar volume such as pillars marked in the green bounding boxes. Most significantly, it can capture ceilings and floors that are almost unperceivable from the original radar volume. Specifically, we see a regression in depth value as shown in the region of radar depth map

Table 5.1: Depth Evaluation Metrics

Abs rel: $\frac{1}{\lvert T\rvert}\sum_{y\in T}\frac{\lvert\tilde{y}-y^*\rvert}{y^*}$	RMSE: $\sqrt{\frac{1}{\lvert T\rvert}\sum_{y\in T}\Vert\tilde{y}-y^*\Vert^2}$
Sqr rel: $\frac{1}{\lvert T\rvert}\sum_{y\in T}\frac{\Vert\tilde{y}-y^*\Vert^2}{y^*}$	Thr: $\% \; y \ni \max(\frac{\tilde{y}}{y^*}, \frac{y^*}{\tilde{y}}) = \delta < thr$

Table 5.2: Quantitative Results on Depth Estimation

	Error (↓), Ours / CFAR					
	Abs Rel		Sqr Rel		RMSE	
EC 0	0.2057	0.2991	0.3197	0.8009	1.1216	1.5249
EC 1	0.2097	0.3785	0.2878	1.1093	0.9916	1.6646
EC 4	0.2407	0.3326	0.3895	0.8231	1.2138	1.5922
Arpg 0	0.2646	0.4135	0.4266	1.0858	1.3312	1.8269
Arpg 1	0.2664	0.4135	0.4324	1.1947	1.3222	1.7778
Arpg 2	0.2723	0.4489	0.4362	1.287	1.3391	1.9322
Arpg 3	0.2635	0.3836	0.3989	1.0547	1.2288	1.6997
Arpg 4	0.2595	0.4525	0.4077	1.2592	1.2612	1.8353
	δ Threshold (↑), Ours / CFAR					
	1.25		1.25^2		1.25^3	
EC 0	0.7004	0.6472	0.881	0.7984	0.9491	0.8816
EC 1	0.6996	0.617	0.8929	0.7777	0.9593	0.8562
EC 4	0.6439	0.5636	0.8441	0.7661	0.9299	0.8642
Arpg 0	0.5447	0.4807	0.8084	0.6862	0.92	0.8203
Arpg 1	0.5555	0.5573	0.8043	0.7289	0.9118	0.828
Arpg 2	0.5332	0.4921	0.7864	0.663	0.9098	0.7947
Arpg 3	0.5691	0.5708	0.8196	0.7365	0.9267	0.8371
Arpg 4	0.573	0.4898	0.8246	0.6817	0.924	0.7999

boxed by the elongated blue boxes. When visualized in the spherical coordinates, these decrease in depth value would mark a sharp curve towards the origin, which does not have a significant peak in the radar intensity volume.

We also provide quantitative results in Table 5.2. We use metrics commonly seen in the monocular depth estimation literature to evaluate the depth map results [13]. We provide a brief summarization in Table 5.1. For the metrics, ↑ shows higher is better, and vice versa.

While there lacks an avaliable baseline method, we compare against cell-averaging CFAR target detector. Since said method is sparse, we only compute the error when there are available measurements. Our method outperforms the baseline method across the board. Between the different sequences, *arpg_lab* score lower overall performance than *ec_hallways*. This is expected since there could be overlaps between the training sequence,

Table 5.3: Quantitative Results on Pixel Classification

	F-Score (↑)		
	Seg	Pho.Conf	std
EC 0	0.7893	0.6294	0.4180
EC 1	0.8311	0.6655	0.3963
EC 4	0.7957	0.6238	0.4088
Arpg 0	0.8204	0.6466	0.4924
Arpg 1	0.8237	0.6513	0.4688
Arpg 2	0.8218	0.6346	0.4832
Arpg 3	0.8314	0.6542	0.4853
Arpg 4	0.8350	0.6716	0.4765

ec_hallways 2, and the rest of *ec_hallways* sequences. However, since *arpg_lab* catches up to the performance of *ec_hallways* when threshold δ value increases from 1.25 to 1.25^2, the degradation in performance is relatively local.

5.3 Out-of-Range Classification

In this section, we compare the quantitative results for the out-of-range pixel classification. This process is analogous to the depth map filtering process in many learning-based multi-view reconstruction methods. We compare our method against the photometric confidence method described in [56], as well as standard deviation for a conventional statistical measure. F-score is used to evaluate the performance. The quantitative results are summarized in Table 5.3. Our 3D convolution based method achieved improved F-scores due to its ability to utilize local information to determine if a pixel is truly out of range.

5.4 Failure Cases

Fig. 5.2 shows typical failure cases of our method. In the red-bounding box, LiDAR measurements penetrated through glass and registered the wall behind the glass, while radar has its measurements absorbed by the glass, resulting in a noisy output that cannot be rejected through the classification model since the output does not match the typical out-of-range characteristics. While failing to register the glass presents a potential danger for navigation tasks from the LiDAR side, radar also fails to detect the glass with clarity. This issue demonstrates the downsides of using LiDAR as the reference depth estimation. It would be an interesting future work to enable the current system to understand different

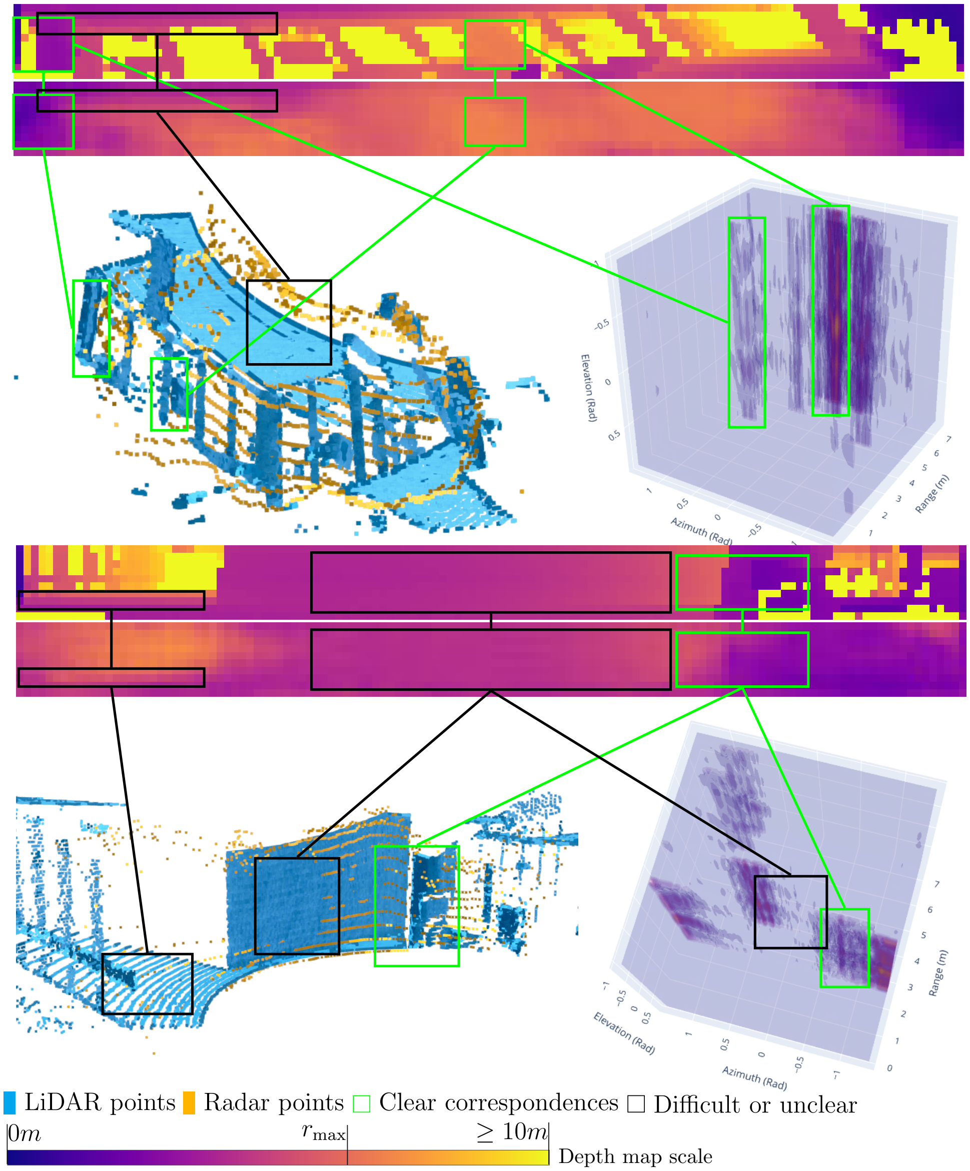


Figure 5.1: Qualitative performance on indoor scenarios. The first row shows LiDAR depth maps, the second row shows inferred radar depth maps. Brighter is farther, $r_{\max} \approx 7.58m$. Note that out-of-range pixels are not masked for visual clarity. The third row contains unprojected points and raw radar intensity volume. In the unprojected points figure, LiDAR points are colored blue, radar points are colored yellow. The original density of LiDAR points are used for better visualization of the scenario. In the radar intensity volume, brighter indicates higher intensity. Bounding boxes indicates correspondence between images: green bounding boxes show estimations that are visible in the raw radar volume; black bounding boxes indicate environment features that are difficult to perceive in the raw radar volume. The figure shows that our method is able to generate visually accurate results, and capture floors and ceilings that are barely visible in the original intensity volume.

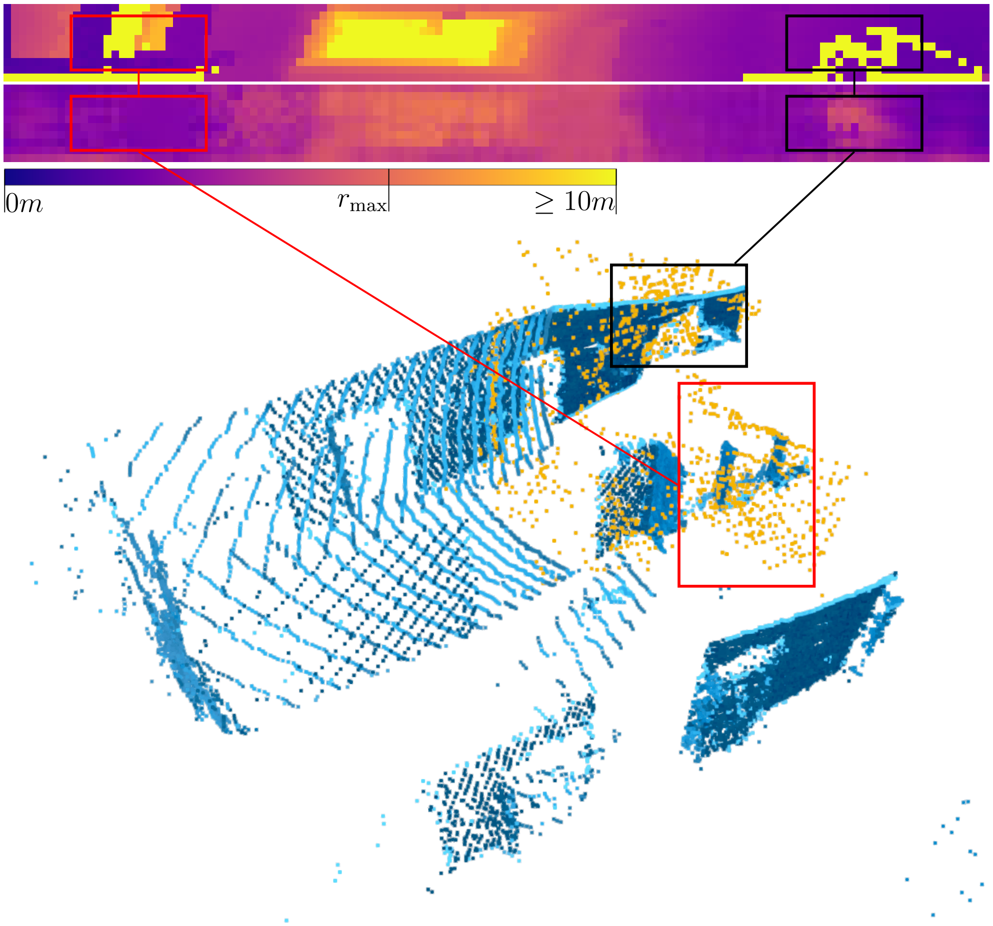


Figure 5.2: Examples where the system performed poorly. From top to bottom: LiDAR depth map, radar depth map, and depth map unprojected into 3D space. In the unprojected points figure, LiDAR points are painted blue and unprojected radar points are painted yellow. Connected boxes show correspondence. The red and black bounding box shows situations where our method fails. The yellow structure inside the black bounding box in LiDAR depth map is from dangling wires from the sensor rig. Looking from other view points, the occluded part is a door.

material properties as [37] did. The blue bounding box is a situation where our methods generated significantly noisier estimates. These situations typically happen when viewing doors from close-range. We suspect it is caused by a combination of noisy close range measurements and the complex reflection path formed by the angles of door frames when in close range.

Chapter 6

Downstream Robotics Tasks using Learned Frontend

In this chapter, we demonstrate the strength of our learned frontend beyond the simple cost metric with respect to ground truth. We show three downstream tasks that are dependent on a reliable frontend. In Section 6.1, we show that our method generates visually salient and consistent occupancy mapping results when given ground truth sensor poses. In Section 6.2, we show that using our learned depth, it is possible to index into the doppler measurements and estimate body frame velocity up to a certain accuracy, and lastly in Section 6.3, we will show that with an existing motion estimate present, it is possible to perform SLAM.

6.1 Occupancy Mapping

In this section, we provide qualitative results for the overall depth map estimation through 3D occupancy mapping using ground truth poses provided in the dataset. Occupancy mapping is performed using OctoMap [19] with cell resolution $0.1m$ and default parameters for occupancy updates. The ground truth is constructed using the LiDAR depth maps created earlier for depth estimation supervision. We also compared our method to the classical CFAR detector. To adapt to the task of mapping, only the first peak detected along each beam of direction is taken as the valid detection. We show three mapping results from *ec_hallway* sequences. All of the ceilings of the mapped scene have been removed for better visualization. The result are shown in Fig. 6.1. Our method successfully captures a vast majority of the structural features, with limited coverage of floors. Mapping using CFAR also results in traces of structural geometry, however the map is much noisier and

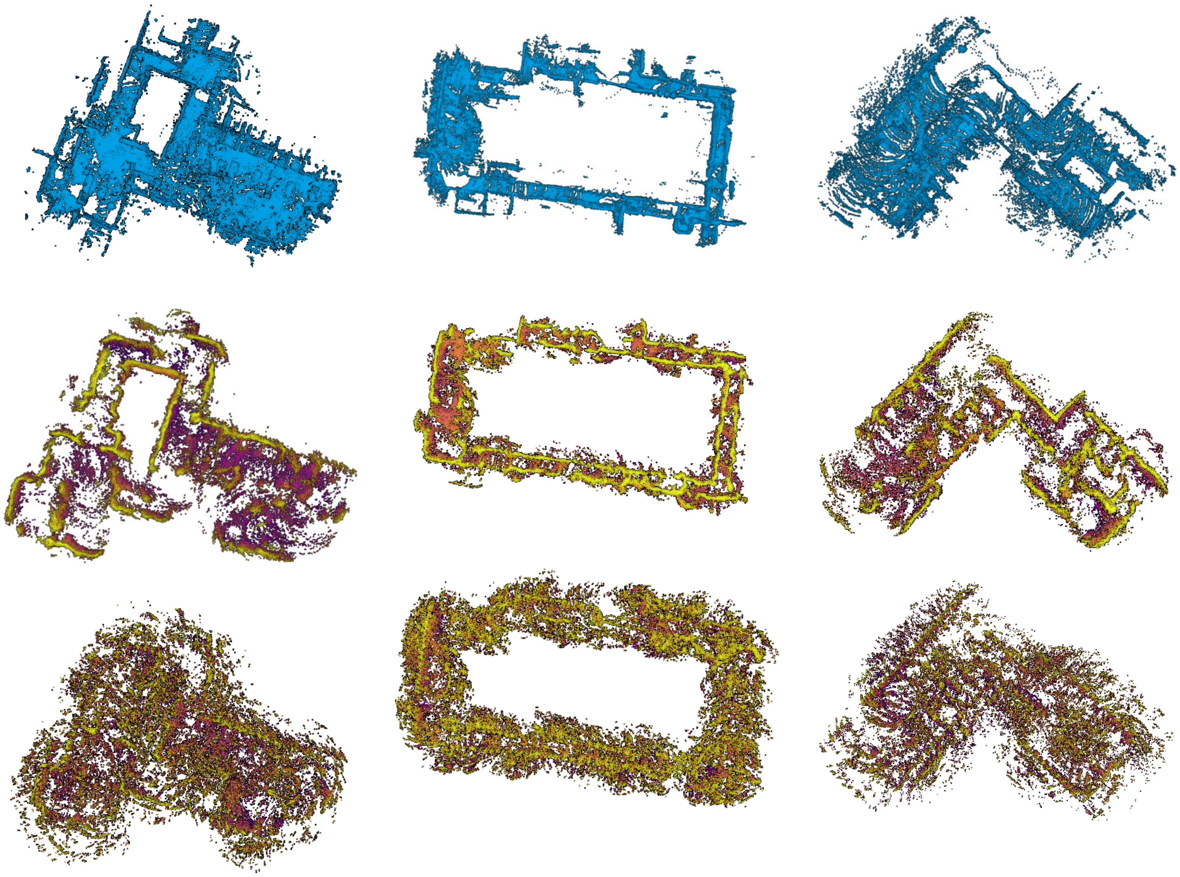


Figure 6.1: From top to bottom: Mapping using LiDAR depth map limited to radar FoV; Mapping using inferred depth estimation of radar; Mapping using CFAR but only the first peak along each beam of direction. All of the maps have ceiling removed naïvely by thresholding over $0.8m$; the floors exist at around $-0.8m$. For the radar maps, darker denotes lower z value. Therefore brighter color usually denotes walls, and large portion of dark region indicates floors. Mapping using CFAR results in noisier maps due to the sparse nature of the detector and the false positive detections that appear independent of adjacent structures.

less usable for robot navigation tasks. Additionally, CFAR has a hard time identifying the correct elevation for a target, which resulted in incorrect spherical surfaces directly under the xy-plane of the map.

6.2 Body Frame Velocity Estimation

Additionally, we show that our learning based depth estimation captures, to some degree, the "real" geometry by estimating the body velocity through indexing depth map into the velocity volume provided in the 4D radar data.

For the radar measurement $\mathbf{V}_v \in \mathbb{R}^{|\mathbf{r}|\times|\boldsymbol{\theta}|\times|\boldsymbol{\phi}|}$ where $\mathbf{V}_v(r, \theta, \phi)$ measures the velocity of the target relative to the sensor along the beam that crosses the sensor origin, with orientation defined by (θ, ϕ). Note that the velocity volume contains noise large enough that it cannot be used to filter invalid voxels in $\mathbf{V}_I$ even when given a good body velocity estimate. However, when given an estimate of sensor frame velocity $\mathbf{v}_s$, the error can be calculated as the following per [30], where $\mathbf{p}_v$ are valid radar detections:

$$e = \sum_{\{r,\theta,\phi\} \in \mathbf{p}_v} \mathbf{V}_v(r,\theta,\phi) + \mathbf{v}_s^\top \left(\frac{\mathbf{t}(r,\theta,\phi)}{||\mathbf{t}(r,\theta,\phi)||} \right) \tag{6.1}$$

$$\mathbf{t}(r,\theta,\phi) = [x,y,z]^\top = r \begin{bmatrix} \cos(\phi)\cos(\theta) \\ \cos(\phi)\sin(\theta) \\ \sin(\phi) \end{bmatrix}$$

Without loss for generality, we assume that there are enough measurements for non-degeneracy. We directly solve for sensor frame velocity through

$$\tilde{\mathbf{v}}_s = \underset{\mathbf{v} \in \mathbb{R}^3}{\arg\min} \sum_{\{r,\theta,\phi\} \in \mathbf{p}_v} \mathbf{V}_v(r,\theta,\phi) + \mathbf{v}^\top \left(\frac{\mathbf{t}(r,\theta,\phi)}{||\mathbf{t}(r,\theta,\phi)||} \right), \tag{6.2}$$

in the form of $\arg\min_x ||Ax - b||^2$ and the covariance is calculated as:

$$\Sigma = \left(A'^\top A' / ||\mathbf{p}_v|| \right)^{-1}, A' = A/(v_{\text{resoln}}/\sqrt{12}). \tag{6.3}$$

We can now assemble $\mathbf{p}_v$ by bitwise masking $\mathbf{I}_r$ with $\mathbf{I}_c$ and calculate the sensor frame velocity. We compare radar-inferred body frame velocity with inferred ground truth velocity through RMSE in Table 6.1.

In all of the sequences, the primary motion is around $1.2m/s$ in the y-axis with fluctuations. Radar experiences the largest error along the x-axis due to the poorer angular resolution as it gets closer to the lateral axis. We argue that since velocity is not part of the learning problem formulation, yet the velocity obtained from the learning outcome closely matches the true body velocity, it implies there exists some degree of learning of true geometry in obtaining dense radar measurements.

6.3 SLAM through Structured Surfaces

When performing perception tasks through artificial environments, one can typically expect the presence of large planes that conforms to either orthogonal or parallel relationships.

Table 6.1: RMSE for Body Frame Velocity Estimation

	RMSE (m/s) $(\downarrow)$		
	v_x	v_y	v_z
EC 0	0.5674	0.1796	0.2623
EC 1	0.5915	0.2525	0.3406
EC 4	0.4923	0.1838	0.2851
Arpg 0	0.4717	0.2114	0.3013
Arpg 1	0.3798	0.2274	0.2696
Arpg 2	0.3477	0.2163	0.255
Arpg 3	0.363	0.3646	0.266
Arpg 4	0.5057	0.2852	0.2774

SLAM with planes has previously been explored in the context of RGB-D and Inertial systems such as [20, 42], and has achieved improved reconstruction results when there is no loop closure constraint to correct the accumulated drift in the system.

In the context of radar, planes are easier to be detected. Regular-sized objects might suffer a smaller radar-cross-section area when angled away from the radar, and therefore a chance of being undetected by the radar system. Planes, especially structural planes, in artificial environments span a much larger area, and naturally have a larger radar-cross-section area. In the raw data, we found that planes are almost the only type of feature that can be consistently distinguished by human eyes. These planes have been preserved in our learned depth map, and we were able to achieve consistent plane detection using a popular open source codebase for point clouds that are not dedicated for radar data.

This observation prompted the following evaluation of utilizing radar for SLAM with assistance from an external source of motion estimate. We will first introduce the parameterization of 3D planes in Section 6.3.1. We will then outline the setup for the SLAM system we used for evaluation in Section 6.3.2. Section 6.3.3 contains the qualitative results.

6.3.1 3D Plane Parameterization

We adopt the plane parameterization used in [24]. A 3D plane is represented as a homogeneous vector in projective space:

$$\boldsymbol{\pi} = (\pi_1, \pi_2, \pi_3, \pi_4)^\top \in \mathbb{P}^3, \tag{6.4}$$

and a homogeneous point $\mathbf{p} = (p_1, p_2, p_3, p_4)^\top \in \mathbb{P}^3$ lies on the plane iff:

$$\boldsymbol{\pi}^\top \mathbf{p} = 0. \tag{6.5}$$

Mapping into $\mathbb{R}^3$ where $p_4 = 1$ produces the following form:

$$\mathbf{n}^\top \mathbf{p}^{xyz} = d, \tag{6.6}$$

where $\mathbf{n}$ is the normal vector of the plane:

$$\mathbf{n} = \frac{(\pi_1, \pi_2, \pi_3)^\top}{||(\pi_1, \pi_2, \pi_3)||}, \tag{6.7}$$

and d is the distance from the origin:

$$d = \frac{-\pi_4}{||(\pi_1, \pi_2, \pi_3)||}. \tag{6.8}$$

This homogeneous plane representation can further be transformed between frames using the following relationship:

$$\boldsymbol{\pi}_g = T_{gx}^{-\top} \boldsymbol{\pi}_x. \tag{6.9}$$

Plane Representation in Optimization

However homogeneous plane representation is over-parameterized and needs to be handled carefully during optimization. Planes only require three parameters to be defined: two angles, and the orthogonal distance from the origin. Only using those three parameters results in singularities similar to those encountered by using Euler angles for 3D orientation. Directly using over parameterization would result in a rank-deficient information matrix that cannot be inverted, which is essential for Gauss-Newton type of optimization. Regularization can mitigate the issues, however would result in slower convergence.

Kaess et al. presented a way for minimal representation by restricting the ambiguity in the homogeneous representation: normalizing the vector $\mathbf{p}$ to lie on the unit sphere of $\mathbb{R}^4$. Take two examples: $(0, 0, 0, 1)^\top$ represents the north pole of the unit sphere, and corresponds to the plane at infinity, $(1, 0, 0, 0)^\top$ is a point on the equator and represents a plane that goes through the origin. Similar to how Quaternions double cover $SO(3)$, negative $\mathbf{p}$ leads to the same plane representation. We can restrict $\mathbf{p}_4 \geq 0$ to obtain unique representation.

There are many options to restrict the optimization on the unit sphere. One solution is to use a minimal representation to update the plane during optimization. A normalized homogeneous plane parameterization can be identified with S^3, and therefore it is possible to simply treat the plane representation as a quaternion, and use the exponential map to update planes during optimization.

Relative Formulation of Planes in SLAM

We use the relative plane formulation as noted in [24]. Planes are now expressed relative to the first pose that they are observed, resulting in the following cost function:

$$
\begin{aligned}
c(\mathbf{x}_b, \mathbf{x}_c, \boldsymbol{\pi}) &= \left|\left|\left(T_{gb}^{-1} T_{gc}\right)^{-\top} \boldsymbol{\pi} \ominus \tilde{\boldsymbol{\pi}}_c\right|\right|_{\Sigma}^{2} && (6.10)\\
&= \left|\left|\log\left(q\left(\left(T_{gb}^{-1} T_{gc}\right)^{-\top} \boldsymbol{\pi}\right)^{-1} q\left(\tilde{\boldsymbol{\pi}}_c\right)\right)\right|\right|_{\Sigma}^{2} && (6.11)
\end{aligned}
$$

between $\mathbf{x}_b$ the first pose this plane is observed, $\mathbf{x}_c$ the current pose, observation of plane $\tilde{\boldsymbol{\pi}}_c$ in $\mathbf{x}_c$, and the plane parameterization. We predict the plane measurement using equation 6.9, calculate the S^3 difference by quaternion multiplication, and maps the difference into tangent space $\mathbb{R}^3$ using Logmap.

Similarly, structural constraints can be expressed as follows:

$$
\begin{aligned}
c_{\perp}(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \mathbf{x}_1, \mathbf{x}_2) &= \left|\left|R_{w1}^{-1} R_{w2} \mathbf{n}_1^{\top} \mathbf{n}_2\right|\right|_{\Sigma}^{2}, && (6.12)\\
c_{\parallel}(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \mathbf{x}_1, \mathbf{x}_2) &= \left|\left|R_{w1}^{-1} R_{w2} \mathbf{n}_1 \times \mathbf{n}_2\right|\right|_{\Sigma}^{2}, && (6.13)
\end{aligned}
$$

where $c_{\perp}$ and $c_{\parallel}$ represents the constraints for perpendicular and parallel planes, $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$ represent the constrained planes, while $\mathbf{x}_1$ and $\mathbf{x}_2$ are the base pose for the planes.

This method allows for better convergence by restricting planes into the local graph. Intuitively speaking, odometry drift may happen over long distance mapping, and loop closure can cause sections of trajectory to move. However, the relative position of planes with respect to their observed poses should not be affected.

6.3.2 SLAM System Setup

We construct a simple SLAM system to show that our learned frontend can generate meaningful depth information. Most importantly, we want to show that the results from probabilistic occupancy mapping were not achieved by pushing a minimum collision free circle along the ground truth trajectory. The system has access to the radar and a noisy odometry estimate. The factor graph is setup as shown in Fig. 6.2.

We use the noisy odometry and depth estimation to accumulate a local submap with probabilistic occupancy mapping. We then use the plane segmentation pipeline in Open3D [62], a popular open source point cloud processing framework, to detect the planes in

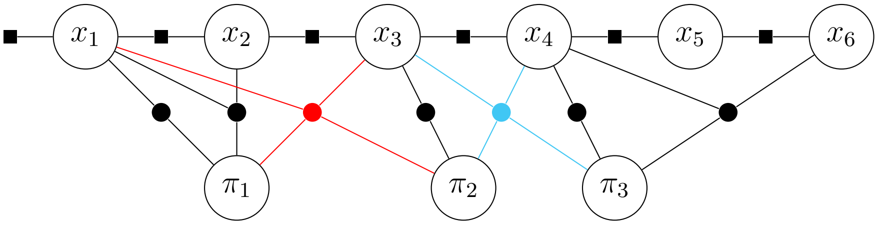


Figure 6.2: Illustration of Factor Graph used in the proposed system. x and π represent robot poses and planes. Square factors are the odometry factors, and circular factors are the plane observation factors. Black plane observation factors are direct co-planar observation, red are orthogonal factors, and blue are parallel factors. Note that due to the relative formulation, subsequent plane observations result in ternary factors that also connect to the first poses that the planes were observed.

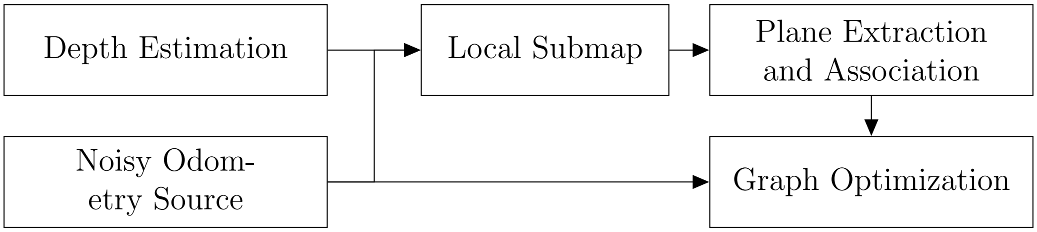


Figure 6.3: System Setup Flowchat.

learned frontend. A flowchart of this system is shown in Fig 6.3. Due to the low number of structural planes present in the observations, we test the association of new planes with all previous planes. The association were judged based on the following two metrics:

1. Angle between the two normal vectors, and
2. Average distances from the center of one plane to the other plane.

We define the center of one plane as the centroid of the inlier points of the plane parameterization. We use Levenberg-Marquardt method [40] to solve the system at every single iteration.

6.3.3 Qualitative Results

Due to the limitation of available radar datasets, we tested our system on the ColoRadar dataset [31]. As this dataset performs 3D motion and lacks odometry information, we simulated the odometry information by adding noises to the ground truth trajectory.

We performed test on the *ec_hallway*#01 sequence of ColoRadar, and we show the qualitative results in Fig 6.4. The system is able to reduce long term drift by enforcing structural constraints and co-planar observations. Previous planar SLAM systems were also successful in reducing short term odometry drift by enforcing planar constraints [20]. We were unable to achieve a similar result since the single frame measurements were noisy,

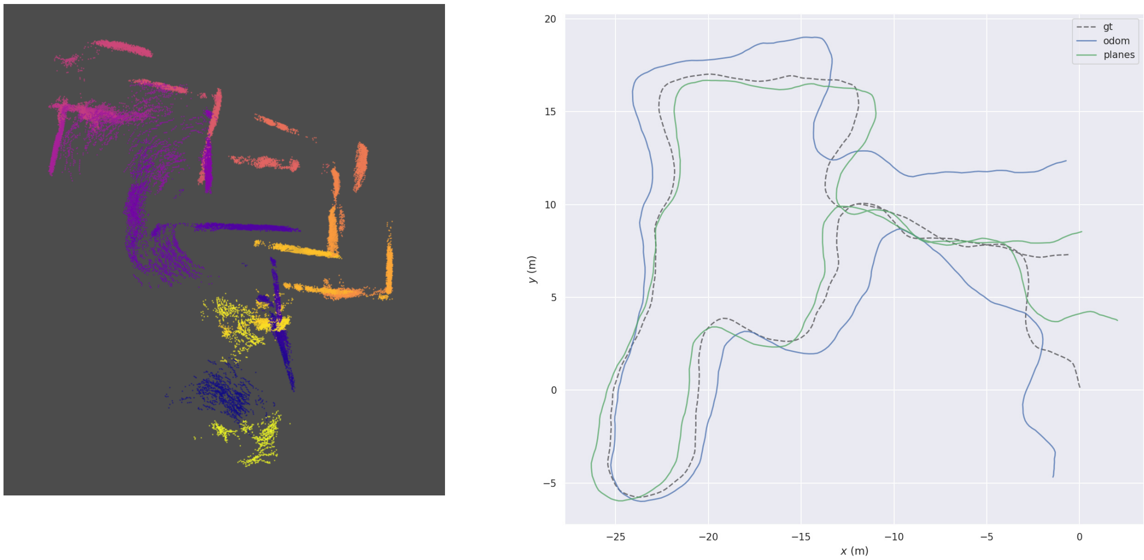


Figure 6.4: Left: Mapping results. Different planes are shown with different colors. Right: Odometry results. Achieved visible drift reduction when compared to odometry (blue).

and can easily throw off the optimization steps.

Additionally, the author wants to note that this system is in no way a robust SLAM system, and will be very sensitive to parameter settings. The work in this section is for demonstrative purposes that the measurements of structural planes are present and consistent.

Chapter 7

Conclusion

7.1 Contribution

In this thesis, we address the challenge of using imaging radar for indoor robot localization and mapping. We propose a novel learning-based approach for estimating depth from raw imaging radar data, and present the system design for collecting a new indoor imaging radar dataset.

In the proposed approach, we formulate the output of radar measurements as cylindrical depth maps with LiDAR supervision. A pixel-wise classification module is created to filter out out-of-range measurements. While raw ADC data from radar measurements still encode much information, such as material property, the decision to represent radar output as depth maps instead of sparse point clouds or dense heat volumes showed the possibility of a unified radar data representation with other popular robotics sensors.

The output of the proposed system is evaluated with downstream robotics tasks. We showcased the output on probabilistic occupancy mapping, body frame velocity estimation, and SLAM with planes.

7.2 Discussions and Future Work

Other than the failure cases we have listed in Section 5.4, there are other areas of improvements and assumptions that deserves a closer look.

While deep-learning methods indeed brought us a better output, it has also made the type of sensor model to use in the factor unclear. This is especially important since our output is still not close enough to the true depth. If we simplify the model and assume a Gaussian distribution, very often it will not work. If the goal is to obtain frame-to-frame motion, then it could be interesting to look at directly learning factors as Baikovitz et al.[2] did for Ground Penetrating Radar.

We have found out that compared to single chip radar, imaging radar, despite achieving denser measurement and better angular resolution, performs worse in terms of velocity estimation. There were significant effort spent in understanding the speed of detected points in the depth output. It could be interesting to look at incorporating the velocity component of radar as part of the loss when determining if a pixel in the radar output depth map is valid. Additionally, 2D radar velocity estimation and outlier rejection has been well-studied [28], however the formulation for 3D radar velocity estimation is still rather simple and straight straightforward. There exists the possibility of a more robust sensor model for incorporating 3D radar velocity into the state estimation framework.

The tremendous potential for using imaging radar in indoor robot navigation should not be ignored. It appears that the primary interest for radar in the automotive industry is object detection. Simply detecting objects is not enough for navigation and mapping tasks in indoor environments. This work has already showed that it is possible to mine out consistent, albeit still very noisy depth measurements from seemingly hard-to-interpret imaging radar data. In the larger scope, imaging radar is a tool that can help enable robust perception and navigation in visually degraded and smoke occluded environments, where traditional sensors typically fail. Future work should aim at leveraging radar-specific characteristics and improve the accuracy of depth estimation, or the encoding of material property in raw ADC data, and perhaps use such measurements as landmark in robot navigation.

Bibliography

[1] Roberto Aldera, Daniele De Martini, Matthew Gadd, and Paul Newman. What Could Go Wrong? Introspective Radar Odometry in Challenging Environments. In *IEEE Intelligent Transportation Systems (ITSC) Conference*, Auckland, New Zealand, October 2019. 1.1

[2] Alexander Baikovitz, Paloma Sodhi, Michael Dille, and Michael Kaess. Ground encoding: Learned factor graph-based models for localizing ground penetrating radar. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5476–5483, 2021. doi: 10.1109/IROS51168.2021.9636764. 7.2

[3] Dan Barnes, Matthew Gadd, Paul Murcutt, Paul Newman, and Ingmar 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. URL `https://arxiv.org/abs/1909.01300`. 3.1

[4] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, 2016. doi: 10.1109/TRO.2016.2624754. 2.1.1

[5] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 3.1

[6] Vincent Casser, Sören Pirk, Reza Mahjourian, and Anelia Angelova. Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. In *AAAI*, 2019. 4.2.2

[7] Sarah H. Cen and Paul Newman. Radar-only ego-motion estimation in difficult settings via graph matching. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 298–304, 2019. doi: 10.1109/ICRA.2019.8793990. 1.1

[8] Andreas Danzer, Thomas Griebel, Martin Bach, and Klaus Dietmayer. 2D car detec-

tion in radar data with pointnets. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, page 61–66. IEEE Press, 2019. doi: 10.1109/ITSC.2019.8917000. URL `https://doi.org/10.1109/ITSC.2019.8917000`. 4.3.1

[9] Frank Dellaert and Michael Kaess. Square root sam: Simultaneous localization and mapping via square root information smoothing. *I. J. Robotic Res.*, 25:1181–1203, 12 2006. doi: 10.1177/0278364906072768. 2.1.1

[10] Frank Dellaert and Michael Kaess. 2017. (document), 2.1, 2.1.1, 2.1.1

[11] Sedat Dogru and Lino Marques. Using radar for grid based indoor mapping. In *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pages 1–6, 2019. doi: 10.1109/ICARSC.2019.8733614. 4.2.3

[12] Wei Dong, Kwonyoung Ryu, Michael Kaess, and Jaesik Park. Revisiting lidar registration and reconstruction: A range image perspective, 2021. URL `https://arxiv.org/abs/2112.02779`. 4.3.1

[13] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 2366–2374, Cambridge, MA, USA, 2014. MIT Press. 5.2

[14] Harold M. Finn and R. S. Johnson. Adaptive detection mode with threshold control as a function of spatially sampled clutter level estimates. *RCA Rev*, 29:414–464, 1968. 4.1

[15] Keegan Garcia, Mingjian Yan, and Alek Purkovic. Robust traffic and intersection monitoring using millimeter wave sensors. Technical report, Texas Instruments, 2018. 1.1, 4.2.2

[16] S. Gasperini, P. Koch, V. Dallabetta, N. Navab, B. Busam, and F. Tombari. R4Dyn: Exploring radar for self-supervised monocular depth estimation of dynamic scenes. In *2021 International Conference on 3D Vision (3DV)*, pages 751–760, Los Alamitos, CA, USA, dec 2021. IEEE Computer Society. doi: 10.1109/3DV53792.2021.00084. URL `https://doi.ieeecomputersociety.org/10.1109/3DV53792.2021.00084`. 4.2.1

[17] Mohammad Tayeb Ghasr, Matthew J. Horst, Matthew R. Dvorsky, and Reza Zoughi. Wideband microwave camera for real-time 3-D imaging. *IEEE Transactions on Antennas and Propagation*, 65(1):258–268, 2017. doi: 10.1109/TAP.2016.2630598. 4.2.1

[18] Junfeng Guan, Sohrab Madani, Suraj Jog, Saurabh Gupta, and Haitham Hassanieh. Through fog high-resolution imaging using millimeter wave radar. In *2020 IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR), pages 11461–11470, 2020. doi: 10.1109/CVPR42600.2020.01148. 4.2.1

[19] Armin Hornung, Kai M. Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 2013. doi: 10.1007/s10514-012-9321-0. URL `https://octomap.github.io`. Software available at `https://octomap.github.io`. 6.1

[20] M. Hsiao, E. Westman, and M. Kaess. Dense planar-inertial SLAM with structural constraints. In *Proc. IEEE Intl. Conf. on Robotics and Automation, ICRA*, pages 6521–6528, Brisbane, Australia, May 2018. 6.3, 6.3.3

[21] Peter J. Huber. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, 35(1):73 – 101, 1964. doi: 10.1214/aoms/1177703732. URL `https://doi.org/10.1214/aoms/1177703732`. 4.3.5

[22] Cesar Iovescu and Sandeep Rao. The fundamentals of millimeter wave radar sensors. Technical report, Texas Instrument. 2.2.1, 2.2.1

[23] Hordur Johannsson, Michael Kaess, Brendan Englot, Franz Hover, and John Leonard. Imaging sonar-aided navigation for autonomous underwater harbor surveillance. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4396–4403, 2010. doi: 10.1109/IROS.2010.5650831. 1.1

[24] M. Kaess. Simultaneous localization and mapping with infinite planes. In *Proc. IEEE Intl. Conf. on Robotics and Automation, ICRA*, pages 4605–4611, Seattle, WA, May 2015. 6.3.1, 6.3.1

[25] Michael Kaess, Ananth Ranganathan, and Frank Dellaert. isam: Incremental smoothing and mapping. *IEEE Transactions on Robotics*, 24(6):1365–1378, 2008. doi: 10.1109/TRO.2008.2006706. 2.1.1

[26] Michael Kaess, Hordur Johannsson, Richard Roberts, Viorela Ila, John Leonard, and Frank Dellaert. isam2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering. In *2011 IEEE International Conference on Robotics and Automation*, pages 3281–3288, 2011. doi: 10.1109/ICRA.2011.5979641. 2.1.1, 2.1.1

[27] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960. 2.1.1

[28] Dominik Kellner, Michael Barjenbruch, Jens Klappstein, Jürgen Dickmann, and Klaus

Dietmayer. Instantaneous ego-motion estimation using doppler radar. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 869–874, 2013. doi: 10.1109/ITSC.2013.6728341. 7.2

[29] Giseop Kim, Yeong Sang Park, Younghun Cho, Jinyong Jeong, and Ayoung Kim. Mulran: Multimodal range dataset for urban place recognition. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Paris, May 2020. 3.1

[30] Andrew Kramer, Carl Stahoviak, Angel Santamaria-Navarro, Ali-akbar Aghamohammadi, and Christoffer Heckman. Radar-inertial ego-velocity estimation for visually degraded environments. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5739–5746, 2020. doi: 10.1109/ICRA40945.2020.9196666. 6.2

[31] Andrew Kramer, Kyle Harlow, Christopher Williams, and Christoffer Heckman. ColoRadar: The direct 3D millimeter wave radar dataset. abs/2103.04510, 2021. URL `https://arxiv.org/abs/2103.04510`. (document), 1.1, 2.2.2, 2.7, 3.1, 5.1, 6.3.3

[32] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 239–248. IEEE, 2016. 4.2.2

[33] Chia-Hung Lin, Yu-Chien Lin, Yue Bai, Wei-Ho Chung, Ta-Sung Lee, and Heikki Huttunen. DL-CFAR: A novel CFAR target detection method based on deep learning. In *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, pages 1–6, 2019. doi: 10.1109/VTCFall.2019.8891420. 4.2.1

[34] Jau-Jr Lin, Yuan-Ping Li, Wei-Chiang Hsu, and Ta-Sung Lee. Design of an fmcw radar baseband signal processing system for automotive application. *SpringerPlus*, 5, 12 2016. doi: 10.1186/s40064-015-1583-5. (document), 2.6

[35] Juan-Ting Lin, Dengxin Dai, and Luc Van Gool. Depth estimation from monocular images and sparse radar data. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10233–10240, 2020. doi: 10.1109/IROS45743.2020.9340998. 4.2.1

[36] Yunfei Long, Daniel Morris, Xiaoming Liu, Marcos Castro, Punarjay Chakravarty, and Praveen Narayanan. Radar-camera pixel depth association for depth completion. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual,*

June 19-25, 2021, pages 12507–12516. Computer Vision Foundation / IEEE, 2021. 4.2.1

[37] 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 *ACM International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2020. 2.2.3, 4.2.3, 5.4

[38] Will Maddern, Geoff Pascoe, Chris Linegar, and Paul Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, 36(1):3–15, 2017. doi: 10.1177/0278364916679498. URL `http://dx.doi.org/10.1177/0278364916679498`. 1.1

[39] 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*, pages 553–558, 2014. doi: 10.1109/ACSSC.2014.7094506. 4.2.1

[40] Jorge J. Moré. The levenberg-marquardt algorithm: Implementation and theory. In G. A. Watson, editor, *Numerical Analysis*, pages 105–116, Berlin, Heidelberg, 1978. Springer Berlin Heidelberg. ISBN 978-3-540-35972-2. 6.3.2

[41] Raúl Mur-Artal, J. M. M. Montiel, and Juan D. Tardós. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015. doi: 10.1109/TRO.2015.2463671. 2.1.2

[42] Viet Nguyen, Ahad Harati, Agostino Martinelli, Roland Siegwart, and Nicola Tomatis. Orthogonal slam: a step toward lightweight indoor autonomous navigation. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5007–5012, 2006. doi: 10.1109/IROS.2006.282527. 6.3

[43] Dong-Hee Paek, Seung-Hyun Kong, and Kevin Tirta Wijaya. K-radar: 4d radar object detection dataset and benchmark for autonomous driving in various weather conditions, 2022. URL `https://arxiv.org/abs/2206.08171`. 3.1

[44] Kun Qian, Zhaoyuan He, and Xinyu Zhang. 3D point cloud generation with millimeter-wave radar. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 4 (4), dec 2020. doi: 10.1145/3432221. URL `https://doi.org/10.1145/3432221`. 4.2.1

[45] Sandeep Rao. Mimo radar. Technical Report SWRA554A, Texas Instrument, May 2017. URL `https://www.ti.com/lit/an/swra554a/swra554a.pdf?ts=`

`1657632465532`. 2.2.2

[46] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, volume 4, pages 850–850, 1987. doi: 10.1109/ROBOT.1987.1087846. 2.1.1

[47] Christian Stetco, Barnaba Ubezio, Stephan Mühlbacher-Karrer, and Hubert Zangl. Radar sensors in collaborative robotics: Fast simulation and experimental validation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10452–10458, 2020. doi: 10.1109/ICRA40945.2020.9197180. 4.2.1

[48] Yue Sun, Zhuoming Huang, Honggang Zhang, Zhi Cao, and Deqiang Xu. 3DRIMR: 3D reconstruction and imaging via mmWave radar based on deep learning. In *2021 IEEE International Performance, Computing, and Communications Conference (IPCCC)*, pages 1–8, 2021. doi: 10.1109/IPCCC51483.2021.9679394. 4.2.1

[49] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005. ISBN 0262201623. 2.1.1

[50] E.A. Wan and R. Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pages 153–158, 2000. doi: 10.1109/ASSPCC.2000.882463. 2.1.1

[51] Claire M. Watts, Patrick Lancaster, Andreas Pedross-Engel, Joshua R. Smith, and Matthew 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)*, pages 4304–4310, 2016. doi: 10.1109/IROS.2016.7759633. 4.2.1

[52] Shunjun Wei, Zichen Zhou, Mou Wang, Jinshan Wei, Shan Liu, Jun Shi, Xiaoling Zhang, and Fan Fan. 3DRIED: A high-resolution 3-D millimeter-wave radar dataset dedicated to imaging and evaluation. *Remote Sensing*, 13(17):3366, Aug 2021. ISSN 2072-4292. doi: 10.3390/rs13173366. URL `http://dx.doi.org/10.3390/rs13173366`. 2.2.3, 4.2.1

[53] Xinshuo Weng, Yunze Man, Dazhi Cheng, Jinhyung Park, Matthew O'Toole, and Kris Kitani. All-In-One Drive: A Large-Scale Comprehensive Perception Dataset with High-Density Long-Range Point Clouds. *arXiv*, 2020. 3.1

[54] Rob Weston, Sarah Cen, Paul Newman, and Ingmar Posner. Probably unknown: Deep inverse sensor modelling radar. 2019. 1.1

[55] Shi Yan, Chenglei Wu, Lizhen Wang, Feng Xu, Liang An, Kaiwen Guo, and Yebin Liu. DDRNet: Depth map denoising and refinement for consumer depth cameras using cascaded CNNs. In *ECCV*, 2018. 4.2.2

[56] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. MVSNet: Depth inference for unstructured multi-view stereo. *European Conference on Computer Vision (ECCV)*, 2018. 4.2.2, 4.3.3, 4.3.4, 5.3

[57] Ji Zhang and Sanjiv Singh. LOAM: lidar odometry and mapping in real-time. In Dieter Fox, Lydia E. Kavraki, and Hanna Kurniawati, editors, *Robotics: Science and Systems X, University of California, Berkeley, USA, July 12-16, 2014*, 2014. doi: 10.15607/RSS.2014.X.007. URL `http://www.roboticsproceedings.org/rss10/p07.html`. 2.1.2

[58] Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. Through-wall human pose estimation using radio signals. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7356–7365, 2018. doi: 10.1109/CVPR.2018.00768. 4.2.1

[59] Mingmin Zhao, Yingcheng Liu, Aniruddh Raghu, Hang Zhao, Tianhong Li, Antonio Torralba, and Dina Katabi. Through-wall human mesh recovery using radio signals. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10112–10121, 2019. doi: 10.1109/ICCV.2019.01021. 4.2.1

[60] Shibo Zhao, Peng Wang, Hengrui Zhang, Zheng Fang, and Sebastian Scherer. TP-TIO: A robust thermal-inertial odometry with deep thermalpoint. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. URL `https://arxiv.org/abs/2012.03455`. 1.1

[61] Lianqing Zheng, Zhixiong Ma, Xichan Zhu, Bin Tan, Sen Li, Kai Long, Weiqi Sun, Sihan Chen, Lu Zhang, Mengyue Wan, Libo Huang, and Jie Bai. Tj4dradset: A 4d radar dataset for autonomous driving, 2022. URL `https://arxiv.org/abs/2204.13483`. 3.1

[62] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018. 6.3.2