Lidar-Aided Blockage Identification For Millimeter-Wave Wireless Communications

by

Tirumalai Vinjamoor Nikhil Srinivas

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved October 2022 by the
Graduate Supervisory Committee:

Ahmed Alkhateeb, Chair
Georgios Trichopoulos
Stefan Myhajlenko

ARIZONA STATE UNIVERSITY

December 2022

ABSTRACT

The recent trends in wireless communication, fueled by the demand for lower latency and higher bandwidth, have caused the migration of users from lower frequencies to higher frequencies, i.e., from 2.5GHz to millimeter wave. However, the migration to higher frequencies has its challenges. The sensitivity to blockages is a key challenge for millimeter wave and terahertz networks in 5G and beyond. Since these networks mainly rely on line-of-sight (LOS) links, sudden link blockages highly threaten the reliability of such networks. Further, when the LOS link is blocked, the network typically needs to hand off the user to another LOS basestation, which may incur critical time latency, especially if a search over a large codebook of narrow beams is needed. A promising way to tackle the reliability and latency challenges lies in enabling proaction in wireless networks. Proaction allows the network to anticipate future blockages, especially dynamic blockages, and initiate user hand-off beforehand. This thesis presents a complete machine learning framework for enabling proaction in wireless networks relying on the multi-modal 3D LiDAR(Light Detection and Ranging) point cloud and position data. In particular, the paper proposes a sensing-aided wireless communication solution that utilizes bimodal machine learning to predict the user link status. This is mainly achieved via a deep learning algorithm that learns from LiDAR point-cloud and position data to distinguish between LOS and NLOS(non line-of-sight) links. The algorithm is evaluated on the multi-modal wireless Communication Dataset DeepSense6G dataset. It is a time-synchronized collection of data from various sensors such as millimeter wave power, position, camera, radar, and LiDAR. Experimental results indicate that the algorithm can accurately predict link status with 87% accuracy. This highlights a promising direction for enabling high reliability and low latency in future wireless networks.

To my family.

ACKNOWLEDGEMENTS

First, I would like to express my deepest gratitude to my advisor, Dr. Ahmed Alkhateeb, for his patience with me and his attentive supervision throughout the duration of my thesis. Without his guidance and support, this thesis would not have been possible. I would also like to thank my supervisory committee members, Dr. Georgios Trichopoulos and Dr. Stefan Myhajlenko for their time and valuable feedback. I would like to thank my colleague and senior Gouranga Charan for being supportive. His valuable input and knowledge helped me overcome some of my misgivings and helped guide my thesis to its completion. I am also grateful for the moral support he offered me during my times of distress and am deeply appreciative that he was always there for me. I also want to thank my lab colleague Tawfik Osman for the valuable discussions and the pleasant working environment that we shared. Last but foremost, I would like to offer my deepest gratitude to my family for always being there for me and providing support through all of my endeavors.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Chapter 1

INTRODUCTION

1.1 Motivation

In recent times, there has been an exponential rise in the demand for higher data rates and low-latency communications to support a host of applications, including but not limited to autonomous driving and Augmented Reality. Millimeter wave (mmWave) and sub-terahertz (sub-THz) communications are dominating in this regard. However, communication in these bands faces several challenges. In order to guarantee sufficient receive power, these systems adopt large antenna arrays and use narrow directive beams at both transmitters and receivers. Further, the mmWave/sub-terahertz wireless communication systems rely on line-of-sight (LOS) links to achieve sufficient receive signal power. Blocking these LOS links by the moving objects in the environment may disconnect the communication session or cause sudden and significant degradation in the link quality. This is due to the high penetration loss of the mmWave/sub-terahertz signals and the much less receive power of the NLOS links compared to the LOS ones [1, 2]. All that highly challenges the reliability and latency of the mmWave/sub-terahertz communication networks.

The reliance of mmWave/sub-THz systems on LOS draws an important parallel with RGB-D cameras and LiDARs, in which visual data (e.g., images and video sequences) and 3D LiDAR point-cloud data capture visible, i.e., LOS, objects. This parallel is very interesting as these systems rely on machine learning and visible objects to perform a variety of tasks depending on object appearance (image-based object de-

tection [3, 4], LiDAR-based object detection [5]) and/or behavior (action recognition [6–8]). In a wireless network, visible objects in the environment are usually the cause of link blockages. Hence, an object detection system powered by machine learning could be utilized to provide a much-needed sense of surrounding to the network; it enables the network to identify objects in its environment and their behavior and utilize that to detect possible blockages. Such capability helps alleviate the strain of link blockages, and as such, this work focuses on developing a *LiDAR-aided link identification* solution for high-frequency wireless networks. In particular, we develop a sensing-aided solution that enables the mmWave basestations to identify whether or not a LOS link is blocked.

1.2 Prior Work: Sensing-Aided Wireless Communication

The use of sensing in wireless communications has been explored with much fervor in recent times. Various examples may be observed through the myriad of academic literature and discussions. Some of the prior work in guiding this thesis are summarized below:

- In [9], RGB images from cameras were utilized in order to enable proactive link-blockage predictions. A deep learning solution was utilized to predict the link blockage status based on the observable sequence of image and beamforming vectors. The approach was utilized in two separate basestations and was used to perform proactive hand-off based on whether a link was blocked from the perspective of either base station. The work achieved substantial gains over solutions that only utilized wireless millimeter wave beam sequences in highly dynamic environments.

- [10] utilized the position data to perform beam selection for the narrow beams.

The positions were obtained from real-world measurement scenarios. The proposed approach was evaluated against existing approaches, and a noticeable improvement in the beam training overhead was observed. Furthermore, the results were evaluated against those obtained from using synthetically generated data sets and a more suitable metric that attempts to quantize an end system objective more suited to real-world operation.

- The literature in [11] proposes an approach that utilizes the LiDAR sensory information to predict the future beam indices corresponding to beamforming vectors. This sensory information was obtained through meticulous real-world measurements. The approach shows promise through the ability to predict the beams without any prior information on the previously optimal beams and without any prior beam calibration. Consequently, the use of the LiDAR sensory information resulted in a significant reduction in the beam training overhead.

- The work performed in [12] utilizes 2D LiDAR point clouds to predict future LOS link blockages. The work involved dividing a simulated scene into various grids and estimating the number of points within each grid block. The data points located at distances greater than 25m away from the LiDAR were discarded. The grid blocks were subsequently recombined to give a 3D histogram which was utilized for NLOS/LOS prediction.

1.3 Summary of performed work

The work described in this thesis was performed over the greater part of 2 years and is split into two halves. The first half was based on developing the multi-modal sensing test bed. The initial work in this thesis includes but is not limited to building the test bed, interfacing the LiDAR to work with the current test bed, data collection

and verification of the data, and post-processing work. The second half of the work involved using the processed data to perform machine learning to identify the transmitter using the available LiDAR point clouds and using the 3D point cloud data to predict the link status, either LOS or NLOS.

1.4 Abbreviations

The abbreviations used in this thesis are summarized in Table 1.1

1.5 Chapter Description

The contributions of this thesis are described in the following chapters as follows:

- Chapter 2 introduces the Real-World Multi-Modal Sensing, and Communication Data set DeepSense6G [13] collected by the Wireless Intelligence Lab, led by the Director of the lab: Dr. Ahmed Alkhateeb. The motivation behind this data set, as well as the test bed utilized for the collection of the said dataset, is described further in this chapter.

- Chapter 3 introduces the target problem, which is LiDAR-based blockage identification and link status prediction. The motivation behind the problem and the selection of the LiDAR data is also included in the chapter. Finally, the chapter concludes with the prescribed approach and its implementation.

Table 1.1: Summary of Abbreviations

BS	Base station
DNN	Deep Neural Network
FOV	Field of View
FMCW	Frequency Modulated Continuous Wave
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
I/Q	In-Phase/Quadrature
JSON	JavaScript Object Notation
LiDAR	Light Detection and Ranging
LOS	Line Of Sight
LRT	Likelihood Ratio Test
MIMO	Multiple Input Multiple Output
mmWave	Millimeter wave
NLOS	Non Line Of Sight
OFDM	Orthogonal Frequency Division Multiplexing
PA	Phased Array
PCAP	Packet Capture
PDF	Probability Density Function
Radar	Radio Detection and Ranging
RGB-D	Red Green Blue-Depth
RTK	Real Time Kinematic
Rx	Receiver
SNR	Signal-to-Noise Ratio
TCP	Transmission Control Protocol
Tx	Transmitter
UDP	User Datagram Protocol
UE	User Equipment
ULA	Uniform Linear Array
USRP	Universal Software Radio Peripheral
UTC	Universal Time Coordinated

Chapter 2

MULTI-MODAL SENSING AND COMMUNICATIONS TEST BED

2.1 Motivation Behind DeepSense

DeepSense 6G is a real-world multi-modal dataset enabling sensing-aided wireless communication applications. It contains co-existing multi-modal data such as vision, mmWave wireless communication, GPS data, LiDAR, and Radar, collected in realistic wireless environments. This data set was collected in realistic environments, in downtown urban environments and suburbs. The dataset is collected at multiple locations, in different weather conditions, and at different times of the day to increase the diversity. Deepsense6G has founded on the premise that sensing improves wireless communication. In other words, machine learning and sensing information can alleviate some of the challenges faced by mmWave/THz wireless communication systems. It aids and enables various applications such as sensing-aided beam prediction and tracking, blockage prediction, resource management, etc.

2.2 Data Collection Test Bed

The data collection was performed using two different types of testbeds: (i) one includes scenarios with a person acting as the transmitter and (ii) scenarios with the transmitter module being in a vehicle). The testbed consists of 2 units: the transmitter and the receiver. Henceforth, the terms receiver unit and basestation refer to the same collection of multi-modal sensors and may be used interchangeably. The various components of the test bed are as follows: The Transmit unit consists of a raspberry

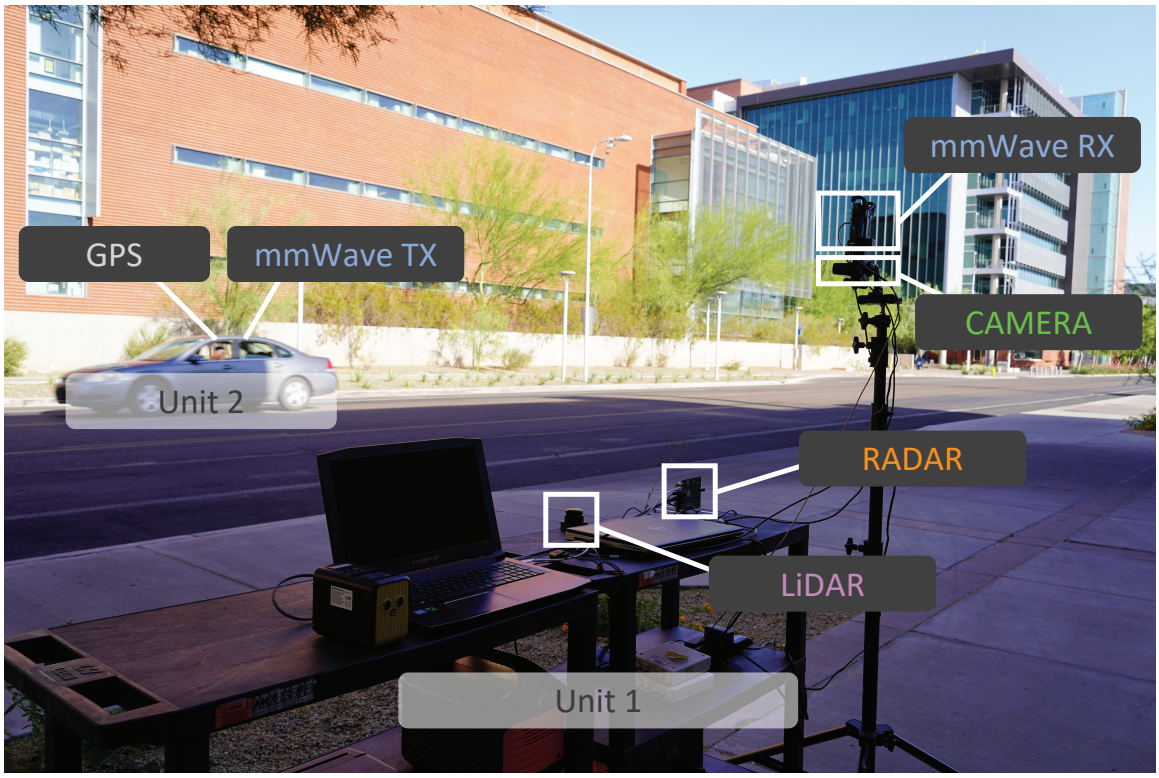
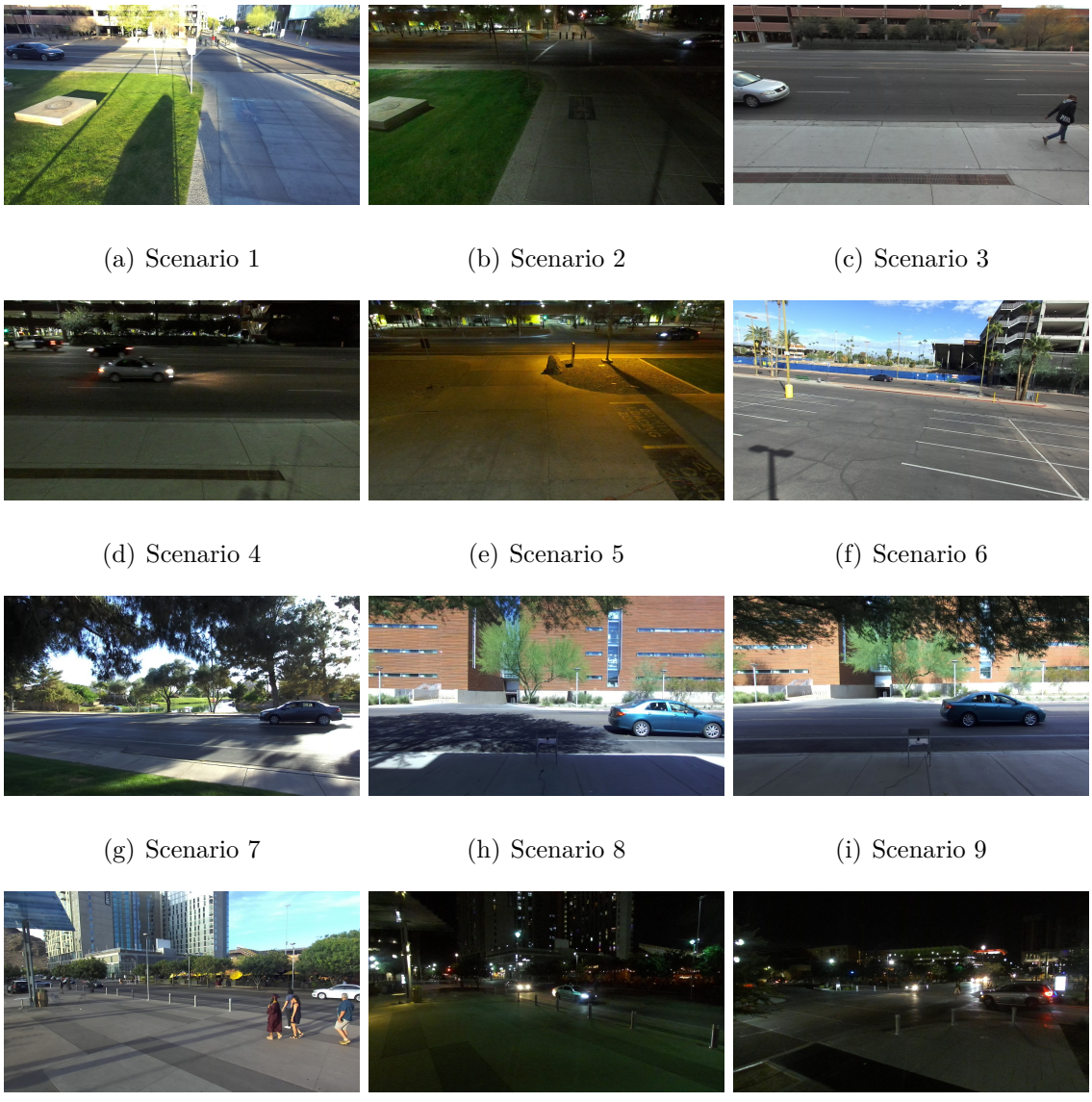


Figure 2.1: This Figure Presents the Deepsense Testbed 1. It Shows the Different Sensing Modalities at the Units 1 and 2.

pi, a USRP, and a Phased array antenna centered at 62.64 GHz(with 64 beam directions). The Phased array used here is a ULA (uniform linear array) with a 60GHz patch antenna, with inbuilt beam steering capabilities by the integrated codebook. The instrument allows full or fractional channels alongside 32-element transmit and receive arrays. The raspberry pi, along with the USRP(which is programmed using GNU radio on Python2 running on the raspberry pi), generates the OFDM symbols fed to the Phased array. The phased array and its drivers are only supported on Python2, hence the need to use an older version of Python here. The OFDM symbols from the USRP output terminal are generated at an intermediate frequency of 1GHz. The Phased array takes these symbols and up-converts them to 62.64GHz, and then transmits the resulting OFDM symbols. The transmitting unit also consists of a



(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

(d) Scenario 4 (e) Scenario 5 (f) Scenario 6

(g) Scenario 7 (h) Scenario 8 (i) Scenario 9

(j) Scenario 32 (k) Scenario 33 (l) Scenario 34

Figure 2.2: This Figure Shows an Example Image Sample from Scenarios 1 - 9 and Scenarios 32- 34

GPS module placed on top of the vehicle and connected to a Windows machine. The GPS module utilized for obtaining the user's position is the RTK Express alongside a GNSS Multi-band antenna, which works for both L1 and L2 frequencies. The GPS is

being operated in Positioning with RTK mode, which provides high-accuracy position measurements (with an accuracy of 1.4cm of the actual position). The GPS antenna is placed on the roof of the vehicle in order to obtain accurate readings and is then initialized using a python script. The GPS coordinates of the vehicle are captured every 100ms, i.e., at a rate of 10 samples per second. The captured data is stored as a text file on the machine.

mmWave Receive Power: The Receiver unit consists of the same Phased array antenna(centered at 62.64Ghz), along with a camera and a GPS receiver to accurately provide the location of the base station. It further possesses a Radar and a LiDAR unit for obtaining out-of-band information about the environment and other objects within the FOV of the receive antenna. We have three different machines at the base station, owing to the large throughput and driver requirements of the various sensors. The different sensors or modalities collect at a different rates. At the receiver antenna, the received symbol is processed and passed through the USRP and gives I/Q samples, where each sample would be a 1x64 vector. The received 2D power matrix is averaged to compute the maximum power in each row. This would be the received beam direction or, more specifically, beam index. This constitutes one of the modalities.

Visual Data: The second modality is collected from an RGB camera. The RGB camera captures an image that consists of RGB and depth. The camera is a 110-degree FOV unit with wide-angle capabilities and an f/1.8 aperture. The camera captures samples at 10Hz, i.e., 10 samples a second. The output of the camera is RGB images of dimension 960×540 pixels.

Position Data: The third modality is GPS, which includes the GPS position of the transmit unit and the GPS position of the basestation.

Radar Data: The fourth modality that is collected is the Radar. The module is

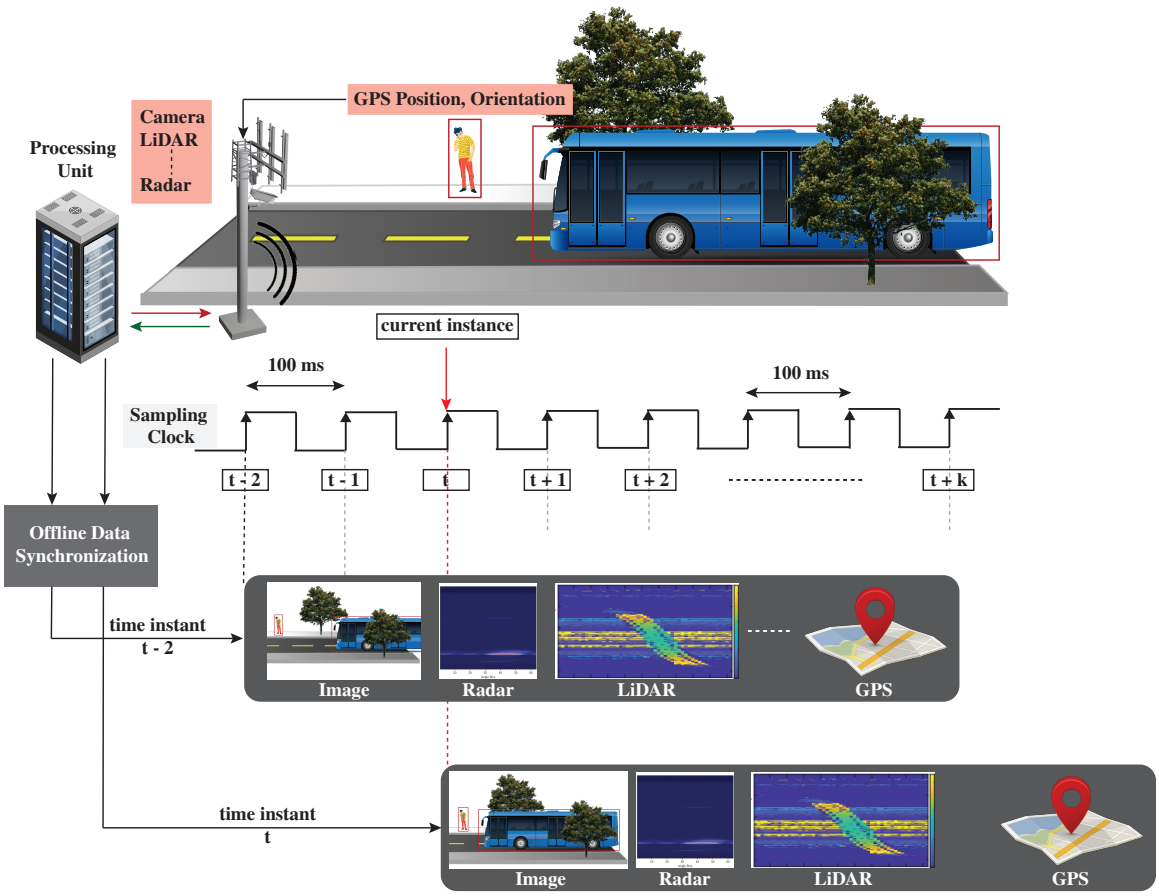


Figure 2.3: The Illustrative Figure Presents the Overall Data Collection Process. It Shows a mmWave Basestation Equipped with a nmWave Array, Camera, Radar, and LiDAR Serving a UE.

a frequency-modulated continuous wave radar (FMCW) with an operating frequency range of 76 to 81 GHz with 4 GHz bandwidth, with a maximum ranger of 100m. The radar is programmed to collect at 10Hz. The entire radar data collection is done through mmWave studio and python. The mmWave studio used a custom Lua script which aids in this data collection. The data collected from the radar consists of 3D complex I/Q radar measurements of (number of Rx antenna) x (samples per chirp) x (chirps per frame).

LiDAR Data: Two types of LiDARs (2D and 3D) are used in the different

DeepSense 6G testbeds. We present the details of both these LiDARs here.

- **2D LiDAR:** The first sensor that was used is a 2-Dimensional LiDAR which collects at 10Hz. The maximum range of this sensor is 14 meters. The sensor utilizes an infrared laser at 905 nm to transmit and then record the reflected light samples. The data collected here is a map of the environment within its current plane. The plane here would refer to an x-y plane with respect to the LiDAR sensor. A simple example here to explain this would be to imagine a sheet of paper whose normal vector lies along the height of the LiDAR sensor. The LiDAR collects coordinate data of moving objects(moving here is an arbitrary term, the objects may be stationary as well) in the polar coordinate system. The data is stored as sets of [r, theta], where r is the distance of an object from the center of the sensor, and theta is the angle the r vector makes with the vector normal to the front of the sensor. The distance 'r' is in meters, and theta is in radians(the range of theta is from -pi to +pi). The LiDAR collects 460 data points, each consisting of one [r, theta] pair every second.

- **3D LiDAR:** The second sensor is a 3-Dimensional LiDAR, which has a 32-channel inbuilt configuration. The word channel refers to the number of vertical beams being used here. The sensor has 1024 horizontal beams, with its FOV in azimuth being 360 degrees. In the vertical direction, the FOV is limited to 54 degrees, being split between 32 beams. The sensor operates in 2 sampling modes. These are the 10Hz and the 20Hz modes. The maximum range of the sensor is 120m; however, if we use the 20Hz sampling rate, we get a 15% reduction in the maximum range. The sensor transmits infrared light at 865nm in all directions. It then receives in all directions and generates the environment map as a point cloud. The sensor data collection is performed through python.

A stringent requirement of Python 3.6+ is required to use the packages in order to run this sensor. The sensor continuously captures and transmits the data to the machine through a UDP connection over Ethernet. Gigabit Ethernet is a must here since the sensor generates a large amount of data. The sensor is configured using the serial number and parameters such as the operating mode. This sensor collects data in the form of large .pcap files(packet capture files). Each .pcap file is over 20 gigabytes in size. The data contained in these files is an environment map, or rather the continuous capture of a changing environment with respect to time. The .pcap files contain a set number of LiDAR frames, which is based on the mode of operation of the sensor. The sensor is operated at 1024x20 mode. This means that we are collecting data at 20Hz from this sensor. A better term here would be collecting data at 20 frames a second. In order to generate our required frames from this large .pcap, MATLAB is utilized. The generated frames are each around 1.5 to 2MB in size. However, a 20 GB .pcap can have more than 40000 frames, meaning that the total size of all the frames combined would exceed 80 GB. Each generated frame consists of several fields.They are timestamp,location,reflectivity,range ,and sensitivity. The timestamp field is used for time synchronization between this sensor and the rest of the modalities. It holds the time(in milliseconds) from the start of the data collection until the instant the frame was collected. The second field(and arguable the most important one) is the location field, which holds the xyz coordinate points of the environment, or rather the point cloud map of the environment. MATLAB handles the conversion from spherical to Cartesian through the Ouster LiDAR toolbox. The point cloud obtained is particularly dense and contains very little noise.). The data captured from all three machines are synchronized using UTC. The UTC may differ from system

to system, but the start time between all the systems is synchronized using a TCP socket connection and transmitting a start message to all the machines simultaneously. The data is captured and stored on the various data collection machines and then undergoes post-processing and manual filtering to obtain the samples in which the transmitter is within the field of view of the basestation. The FOV of the base station is the same as the minimum FOV amongst all the utilized sensors. This is to ensure that the data is consistent within all the sensors. Thus the FOV of the camera is used in this case to decide the FOV of the base station.

2.3 Procedure For Data Collection

The UE/transmitter unit consists of the GPS with a Windows machine, a Phased array antenna centered at 62.64GHz (the Tx antenna), a NI B210 USRP, and a raspberry pi placed inside the vehicle. The GPS is first initialized and starts capturing the vehicle's position along with the UTC every 100ms. The phased array is then initialized to transmit quasi-omni-directionally with a transmit gain of 90dBm. The transmitted waveform is an OFDM waveform at the previously noted center frequency. The vehicle is then driven around the zone of interest. The Tx antenna is oriented in a way such that it always faces the BS. The BS consists of a Phased array antenna centered at 62.64 GHz (the Rx antenna), an RGB camera, a GPS unit, an FMCW radar, and a 3D LiDAR unit are initialized sequentially. The data is collected from all the sensors simultaneously, along with the UTC for each sample. This timestamp is used to synchronize the sensor data during the post-processing phase. The overall data collection process is presented in Figure 2.3. The detailed structure of the 3D LiDAR data is presented next.

2.4 Structure Of The LiDAR Data

The LiDAR maps the environment in two steps. The sensor transmits beams of infrared lasers in all directions. Once the laser is reflected from any object, LiDAR measures the time taken between the transmission and the reception of the beam. It uses this time to calculate the distance where the object is located. The angles at which the LiDAR can receive depend on the configuration of the LiDAR, i.e., the number of vertical and horizontal beams along with the vertical and horizontal FOV. In the case of the 3D LiDAR, the number of vertical beams is 32, with a FOV of 45 degrees. This gives a vertical resolution of 1.406 degrees. The azimuthal FOV is 360 degrees with 1024 beams. This gives a horizontal resolution of 0.351 degrees. The LiDAR converts the received data into LiDAR data packets. Each packet consists of 16 azimuth blocks and is always 12608 bytes in length. The format is as follows:

Table 2.1: LiDAR Data Format

Word	Azimuth Block 0	..	Azimuth Block 15
(Word 0,1)	Timestamp	..	Timestamp
(Word 2[0:15])	Measurement ID	..	Measurement ID
(Word 2[16:31])	Frame ID	..	Frame ID
(Word 3)	Encoder Count	..	Encoder Count
(Word 4,5,6)	Channel 0 Data Block	..	Channel 0 Data Block
(Word 7,8,9)	Channel 1 Data Block	..	Channel 1 Data Block
.	.	.	.
(Word 193,194,195)	Channel 63 Data Block	..	Channel 63 Data Block
(Word 196)	Azimuth Data Block Status	..	Azimuth Data Block Status

Each azimuth blocks consists of the following data:

Table 2.2: Packet Format

Word	Byte 3	Byte 2	Byte 1	Byte 0
(Word 0)	unused[31:24]	range_mm[19:16]	range_mm[15:8]	range_mm[7:0]
(Word 1)	signal_photons[31:24]	signal_photons[23:16]	reflectivity[15:8]	reflectivity[7:0]
(Word 2)	unused[31:24]	unused[23:16]	noise_photons[15:8]	noise_photons[7:0]

- Timestamp [64-bit unsigned int] - timestamp of the measurement in nanoseconds
- Measurement ID [16 bit unsigned int] : a sequentially incrementing azimuth measurement counting from 0 to 511, or 0 to 1023, or 0 to 2047 depending on LiDAR mode.
- Frame ID [16 bit unsigned int] - index of the LiDAR scan. Increments every time the sensor completes a rotation, crossing the zero point of the encoder.
- Encoder Count [32-bit unsigned int] - an azimuth angle as a raw encoder count, starting from 0 with a max value of 90111 - incrementing 44 ticks every azimuth angle in 2048, 88 ticks in 1024, and 176 ticks in 512 modes.
- Data Block [96 bits] - 3 data words for each of the 16 or 64 pixels. The format of the data block is provided
  - Range [32-bit unsigned int - only 20 bits used] - range in millimeters, discretized to the nearest 3 millimeters.
  - Signal Photons [16 bit unsigned int] - signal intensity photons in the signal return measurement are reported
  - Reflectivity [16 bit unsigned int] - sensor signal photon measurements are

scaled based on measured range and sensor sensitivity at that range, providing an indication of target reflectivity.

  - Ambient Noise Photons [16 bit unsigned int] - ambient noise photons in the ambient noise return measurement are reported

- Azimuth Data Block Status [32 bits]- indicates whether the azimuth block contains valid data in its channels' Data Blocks. Good = 0xFFFFFFFF, Bad = 0x0. If the Azimuth Data Block Status is bad (e.g., in the case of column data being dropped), words in the data block will be set to 0x0, but Timestamp, Measurement ID, Frame ID, and Encoder Count will remain valid.

The LiDAR packets are unusable in their current state. They need to be converted into something which can be easily read and understood. The data may be transformed into 3D Cartesian coordinates in the LiDAR coordinate frame. The following data is available within a LiDAR packet:

- encoder count of the azimuth block (η)
- range from the data block of the i-th channel (r)
- beam altitude angles (α)
- beam azimuth angles (β)

The corresponding 3D point can be computed as

$$x = r\cos(\theta)\cos(\phi); \tag{2.1}$$

$$y = -r\sin(\theta)\cos(\phi); \tag{2.2}$$

$$z = r\sin(\phi), \tag{2.3}$$

where $\theta = 2\pi\left(\frac{\eta}{90112} + \frac{\beta[i]}{360}\right)$, $\phi = 2\pi\left(\frac{\alpha[i]}{360}\right)$, and r is the range from the data block of the i-th channel. The range of i here is from 1 to 63. In order to process the LiDAR data,

a .pcap file is opened in MATLAB by using the corresponding .json configuration file from the sensor. The data blocks within the packet are examined and the data within each of the blocks are extracted. The data is converted to co-ordinate points within the LiDAR coordinate system. This system is identical to a spherical co-ordinate system. Finally, the frames are plotted and are saved as corresponding polygon files. The timestamps of each of the frames are also saved to aid in time synchronization.

Chapter 3

3D LiDAR Aided Link Status Identification

3.1 Introduction

Millimeter-wave (mmWave) and sub-terahertz (sub-THz) communications are becoming dominant directions for current and future wireless networks [1, 14]. With their large bandwidths, they have the ability to satisfy the high data rate demands of several applications such as wireless virtual/augmented reality (VR/AR) and autonomous driving. Communication in these bands, however, faces several challenges at both the physical and network layers. One of the key challenges arises from the sensitivity of the high-frequency signals (i.e., mmWave and sub-THz) to blockages [2]. These signals suffer from high penetration loss and attenuation, resulting in strong dips in the received signal-to-noise ratio (SNR) whenever an object is present in-between a basestation and a user. Such dips lead to sudden disruptions of the communication channel, which severely impact the reliability of wireless networks. Re-establishing LOS connection is usually done reactively, which incurs critical network latency [15]. Given all that, high-frequency wireless networks need not only maintain line-of-sight (LOS) connections but also do so proactively, which implies a critical need for a sense of surrounding.

In wireless networks, the primary reason behind LOS link disruptions or blockages, are visible objects. Thus one could identify when such objects would interrupt the link. An object detection neural network utilizing the details captured in a 3D point cloud could provide a much needed sense of the environment to the wireless network,

facilitating the identification of objects by the wireless network and their behavior. Such information could be utilized to detect possible blockages. Such capabilities could alleviate the strain that blockages cause to a link, and thus this work focuses on developing a *LiDAR-aided link identification* solution for high-frequency wireless networks. In particular this work attempts to develop a sensing aided solution which enables the mmWave basestations to identify whether or not a LOS link is blocked.

3.1.1 Prior Works

The problem of LOS link blockage has long been acknowledged as a critical challenge to high-frequency wireless networks [1, 16–18]. In those networks, the quality of service highly deteriorates with link blockages. Therefore, solutions centered around multi-connectivity are a major avenue to handle that problem [17]. For instance, [18] proposes a multi-cell measurement reporting system to keep track of the link quality between a mmWave user and multiple basestations. All basestation in that system feed their measurements to a central unit that takes care of cell selection and scheduling. This system is further studied and tested in [17] under realistic dynamic scenarios. A slightly different look on multi-connectivity is presented in [19, 20]. In [19], the authors propose a few approaches for multi-connectivity, all of which focus on utilizing low-frequency bands (sub-6 GHz) to support the mmWave network. [20], on the other hand, develops a multi-connectivity algorithm that does not only factor in network reliability but also latency. Collectively, the work on multi-connectivity has its promise and elegance, yet it is lacking on two important fronts. First, it is inherently wasteful in terms of resource utilization; multiple basestations schedule resources for one user as a precaution for probable LOS blockages. The other is its reactive nature; the majority of the multi-connectivity algorithms are designed to react to link blockages, not anticipate them.

A new trend in addressing LOS blockages has been developing in recent years, in which the driving power is machine learning [9, 12, 21–25]. The work in [21] studies proactive blockage prediction and hand-off for a single-moving mmWave user in the presence of *stationary* blockages. The proposed solution utilizes observed sequences of mmWave beamforming vectors (beams) and uses a Gated Recurrent Unit (GRU) network to learn beam patterns that proceed link blockages. Again, despite its appeal, it still falls short in meeting the latency and reliability requirements as the sensory data are only expressive of stationary blockages. On a different note, the work in [22] explores a new dimension for blockage prediction in single user communication settings. It proposes a modified residual network [26] that uses visual data to predict stationary blockages. However, like its wireless-data counterparts, it struggles in dealing with complex scenarios with dynamic blockages. Predicting dynamic blockages require more information about these moving blockages in the environment. [9] utilizes the optimal beam indices and the RGB images of the wireless environment to predict dynamic link blockages proactively. However, this solution was based only on synthetic dataset, raising an important question about its practicality, i.e., can this promising solutions be achieved in real-world? Further, visual data is severely impacted by poor weather and lighting conditions making it challenging for these solutions to perform efficiently in real-world scenarios. In [12, 23], in-band mmWave and sub-6GHz based wireless scattering signatures were used to identify/predict the incoming mmWave link blockages. In particular,[23] proposes to use the sub-6GHZ channels to decide whether or not the mmWave LOS link is blocked. These solutions, however, are mainly capable of predicting immediate blockages and are hard to scale to complex/crowded scenarios. Solutions based on radar and 2D LiDAR sensory data were proposed for the first time in [12, 24, 25]. Despite their promising results, easy sensing modality has its advantages and drawbacks. For example, radar

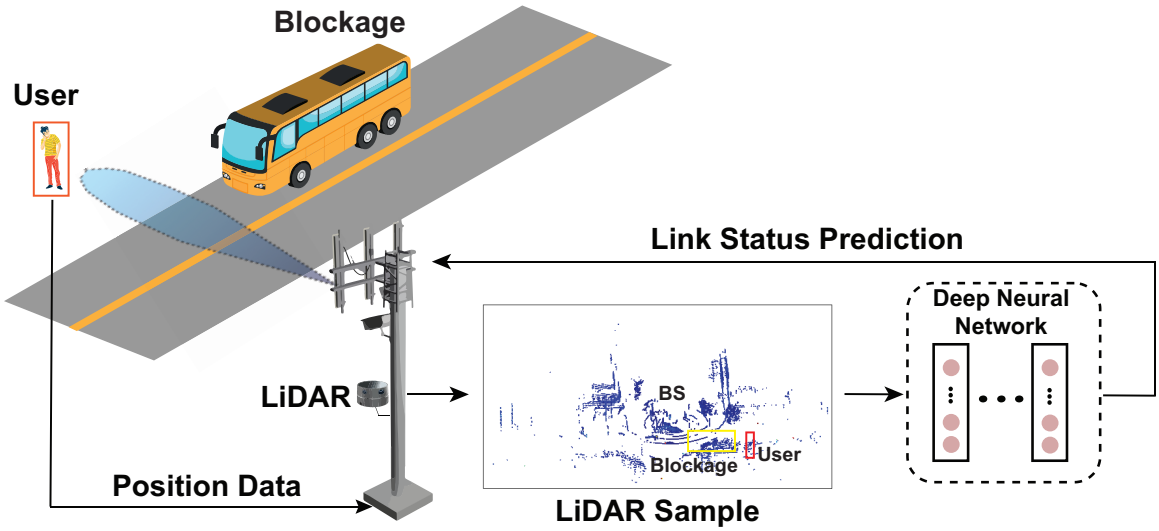


Figure 3.1: The Illustrative Figure Shows a mmWave Basestation Equipped with a mmWave Array and 3D LiDAR Serving Multiple Mobile Users. The Objective of This Work Is to Utilize Position and 3D LiDAR Point-cloud Data to Predict the Link Status of the User.

data are mainly suitable for uncrowded scenarios and 2D LiDAR sensors can only capture limited information, making it difficult for these sensors to capture detailed information of the real-world wireless environment.

3.1.2 Contribution

In [12, 27], the authors proposed a sensing-aided (image and 2D-LiDAR, respectively) machine learning-based solution to predict the LOS link blockage proactively. However, both these approaches were evaluated on real-world scenarios with stationary transmitter and receiver. Therefore, an important question that arises is can we extend these solutions to a more realistic and challenging dynamic scenario with mobile transmitters. In this paper, we attempt to answer this question. As such, this work develops a deep neural network that learns to predict the LOS link status by using the multi-modal 3D-LiDAR point cloud and the position data of the user. The

main contributions of this paper could be summarized in the following few points:

- Formulating the 3D LiDAR-aided blockage identification problem in mmWave/THz wireless networks considering 3D LiDAR point-cloud data collected in the real-world.
- Developing a machine learning approach that is capable of (i) pre-processing the real-world point-cloud data to enhance the link status identification performance, (ii) extracting the relevant features about the scatterers/environment, and (iii) efficiently predicting the user equipment (UE) link status.
- Providing the first real-world evaluation of 3D LiDAR-aided blockage prediction based on our large-scale dataset, DeepSense 6G [13], that consists of co-existing multi-modal sensing and wireless communication data.

3.2 System and Channel Model

In order to illustrate the potential of deep learning and additional sensing-data in mitigating the link blockage problem, this work considers a high-frequency communication network where basestations utilize 3D LiDARs to capture relevant information from the wireless environment. The following two subsections provide a detailed description of the system and wireless channel models adopted in this work.

3.2.1 System model

The communication system considers a small-cell mmWave basestation deployed in an outdoor environment. The basestation is equipped with a uniform linear array (ULA) with M elements and a 3D LiDAR sensor. For practicality [23], the basestation is assumed to employ analog-only architecture with a single RF chain and M phase shifters. As a result of this architecture, the basestation adopts a predefined

beamforming codebook $\boldsymbol{\mathcal{F}} = \{\mathbf{f}_q\}_{q=1}^{Q}$, where $\mathbf{f}_q \in \mathbb{C}^{M\times 1}$ and Q is the total number of beamforming vectors. The choice for $\boldsymbol{\mathcal{F}}$ in this paper is a beam-steering codebook that follows from the choice of the antenna array, i.e., a ULA. For such a codebook, each beamforming vector $\mathbf{f}_q,\ \forall q \in \{1,\ldots,Q\}$ is given by

$$\mathbf{f}_q = \frac{1}{\sqrt{M}}\left[1, e^{j\frac{2\pi}{\lambda}d\sin(\phi_q)}, \ldots, e^{j(M-1)\frac{2\pi}{\lambda}d\cos(\phi_q)}\right]^T, \tag{3.1}$$

where λ is the wavelength, and $\phi_q \in \{\frac{2\pi q}{Q}\}_{q=0}^{Q-1}$ is a uniform quantization of the azimuth angle with an integer step of q. The communication system in this work adopts OFDM with a cyclic prefix of length D and K subcarriers. For any mmWave user in the wireless environment, its received downlink signal is given by

$$y_{u,k} = \mathbf{h}_{u,k}^T \mathbf{f}_q x + n_k, \tag{3.2}$$

where $y_{u,k} \in \mathbb{C}$ is the received signal of the uth user at the kth subcarrier, $\mathbf{h}_{u,k} \in \mathbb{C}^{M\times 1}$ is the channel between the BS and the uth user at the kth subcarrier, $x \in \mathbb{C}$ is a transmitted complex symbol that satisfies the following constraint $\mathbb{E}\left[|x|^2\right] = P$, where P is a power budget per symbol, and finally n_k is a noise sample drawn from a complex Gaussian distribution $\mathcal{N}_{\mathbb{C}}(0,\sigma^2)$.

3.2.2 Channel model

The channel model adopted throughout this paper is a geometric mmWave channel model with L clusters. This model captures the limited scattering property of the mmWave band [28, 29]. The channel vector of the uth user at the kth subcarrier is given by

$$\mathbf{h}_{u,k} = \sum_{d=0}^{D-1}\sum_{\ell=1}^{L} \alpha_\ell e^{-j\frac{2\pi k}{K}d} p\left(dT_{\mathrm{S}} - \tau_\ell\right)\mathbf{a}\left(\theta_\ell,\phi_\ell\right), \tag{3.3}$$

where L is number of channel paths, $\alpha_\ell, \tau_\ell, \theta_\ell, \phi_\ell$ are the path gains (including the path-loss), the delay, the azimuth angle of arrival, and the elevation angle of arrival,

respectively, of the ℓth channel path. T_{S} represents the sampling time while D denotes the cyclic prefix length (assuming that the maximum delay is less than DT_{S}).

Modeling signal blockage: (3.3) is generic in the sense it can describe the channel for both LOS and NLOS users. However, for the sake of clarity in this paper, the equation will be adjusted to convey an explicit definition of LOS and NLOS channels. Let $\mathbf{w} = [w_1, \ldots, w_L]^T$ where $w_\ell \in \{0, 1\},\ \forall \ell \in \{1, \ldots, L\}$ represents a vector of binary variables (indicators). Now, without loss of generality, let the first path in (3.3), i.e., $\ell = 1$, be defined as the LOS path going directly from a user to the BS. The geometric channel model in (3.3) could be re-written as

$$\mathbf{h}_{u,k} = \sum_{d=0}^{D-1} \sum_{\ell=1}^{L} w_\ell^{(u)} \left[\alpha_\ell e^{-\jmath \frac{2\pi k}{K} d} p\left(dT_{\mathrm{S}} - \tau_\ell\right) \mathbf{a}\left(\theta_\ell, \phi_\ell\right) \right]. \tag{3.4}$$

to model both a LOS and a NLOS channels. More to the point, a user u is considered NLOS when its channel vector $\mathbf{h}_{u,k}$ is accompanied with a binary vector $\mathbf{w}$ that has $w_1^{(u)} = 0$, and it is LOS when $w_1^{(u)} = 1$.

3.3 Problem Formulation

A significant problem faced in high-frequency wireless networks is the LOS link blockages. The severity of these problem mostly revolves around the mixed-dynamics in the wireless environment, i.e., it is characterized by a mixture of dynamic and stationary objects. Developing a solution is tightly linked to equipping the wireless network with a sense of its surroundings; such sense transforms the network from being *reactive* to its environment to being *proactive* in it. This simply means having a network able to predict incoming blockages and initiate hand-off procedures beforehand. With that in mind, this work attempts to utilize machine learning and a fusion of 3D LiDAR point cloud and position data to enable that sense of surrounding in a wireless network. In this work, we primarily focus on predicting whether or not the

mmWave LOS link is blocked. The objective is to observe a multi-modal data sample of LiDAR and position at a basestation and use that data sample to predict whether the user is LOS or not. Such prediction task is made possible by two important facts: (i) The LiDAR point-cloud data, in general, is rich with information about the scene they depict, e.g., the type of objects, their relative positions to one another; and (ii) the position data provides the directional information. The following two subsections will lay the groundwork for the proposed solutions by providing formal definitions for the link status identification problem.

3.3.1 Why LiDAR?

LiDAR utilizes beams of infra-red lasers are used to determine the location and the shape of an object. The wavelength of the emitted radiation lies between 750 nm and 1.1µm. This constitutes the near-infra-red spectrum. 2D LiDAR uses a wavelength of 905 nm, whereas 3D LiDAR uses a wavelength of 865nm. The larger optical aperture of 3D LiDAR improves the sensor's performance in dusty environments due to only partial attenuation of the laser rather than complete obscuring of the laser. The selected wavelength also improves performance in foggy environments due to the minimum laser absorption by water vapor. LiDAR has an innate ability to capture rich information about the spatial structure of the environment. In short, it generates a 3-dimensional model of a dynamic environment in the form of a point cloud. This rich information can be utilized to extract features of our region of interest within the environment (the region where the transmitter can be found). The distance and shape of the transmitter are some features that can be extracted from the point clouds. Such information can be leveraged using machine learning. It can be used in applications such as predicting the beam direction (beam indices) or detecting where the transmitter will be based on an arriving sequence of point cloud data.

A good question to ask here is, why not use images to perform the same task? The answer to that question lies in the procurement of the data itself. Images require a camera to be present at the base station. Using cameras, in some situations, opens up a slew of problems, especially regarding privacy concerns; people are not so inclined as to be recorded with their consent. LiDAR is a way around those problems. The sensors do not capture details of a person's face, which instead looks like a cluster of random points in a 3D point cloud. The other advantage of LiDAR is the ability to capture information even when there is no visible light. The same cannot be said of cameras. LiDAR could even perform much better in low-light scenarios, as the interference from sunlight is vastly reduced.

3.3.2 Link Status Identification

The primary objective of this paper is to utilize a multi-modal sensing data (3D LiDAR point cloud of the wireless environment and position of the UE) and develop a machine learning model that learns to predict whether the UE is LOS or NLOS. Formally, this learning problem could be posed as follows. For any user u in the environment, at any given time instant t, a LiDAR and position data pair is observed. This can be expressed as

$$\mathcal{S}_u = \{(\mathbf{X}_u[t], \mathbf{b}_u[t])\}, \tag{3.5}$$

where $\mathbf{b}_u[t]$ is the ground-truth position of the user consisting of the latitude and longitude information at the tth time instance, $\mathbf{X}_u[t]$ is the LiDAR point cloud data captured at the tth time instance. For robust network operation, the objective is to observe $\mathcal{S}_u$ and predict whether the link is blocked or not. Then, the overall link status of the uth user can be defined as $s_u \in 0, 1$, where 0 indicates a LOS connection and 1 indicates that the link is blocked.

The primary objective is attained using a machine learning model. It is developed

to learn a prediction function $f_\Theta(\mathcal{S})$ that takes in the observed sensing data and produces a prediction on the UE link status $\hat{s} \in \{0, 1\}$. This function is parameterized by a set Θ representing the model parameters and learned from a dataset of labeled samples. To put this in formal terms, let $\mathbb{P}(\mathcal{S}, s)$ represent a joint probability distribution governing the relation between the observed data samples $\mathcal{S}$ and the current link status s in some wireless environment, which reflects the probabilistic nature of link status in the environment. A dataset of independent pairs $\mathcal{D} = \{(\mathcal{S}_u, s_u)\}_{u=1}^{U}$ where $(\mathcal{S}_u, s_u)$ is sampled at random from $\mathbb{P}(\mathcal{S}, s)$—$s_u$ is serving as a *label* for the observed samples $\mathcal{S}_u$. This dataset is then used to train the prediction function $f_\Theta(\mathcal{S})$ such that it maintains high-fidelity predictions for any dataset drawn from $\mathbb{P}(\mathcal{S}, s)$. This could be mathematically expressed as

$$\max_{f_\Theta(\mathcal{S})} \quad \prod_{u=1}^{U} \mathbb{P}(\hat{s}_u = s_u | \mathcal{S}_u), \tag{3.6}$$

where the joint probability in (3.6) is factored out as a result of the independent and identically distributed samples in $\mathcal{D}$. This conveys an implicit assumption that for any user u in the environment, the success probability of $f_\Theta(\mathcal{S}_u)$ predicting s_u only depends on its observed sequence $\mathcal{S}_u$.

3.4 Proposed Solution

This section presents the proposed solution for predicting user link status in a real-wireless environment with multiple candidates. It proposes a novel approach that utilizes bimodal LiDAR and position data in $\mathcal{D}$ to identify the link status. A three-step architecture is proposed for this task. The first step of the proposed architecture relies on DNNs to produce bounding boxes enclosing relevant objects in the scene. It is performed to detect all the probable transmitting objects in the environment. In the second step, the DNN uses position data to predict the probable bounding-

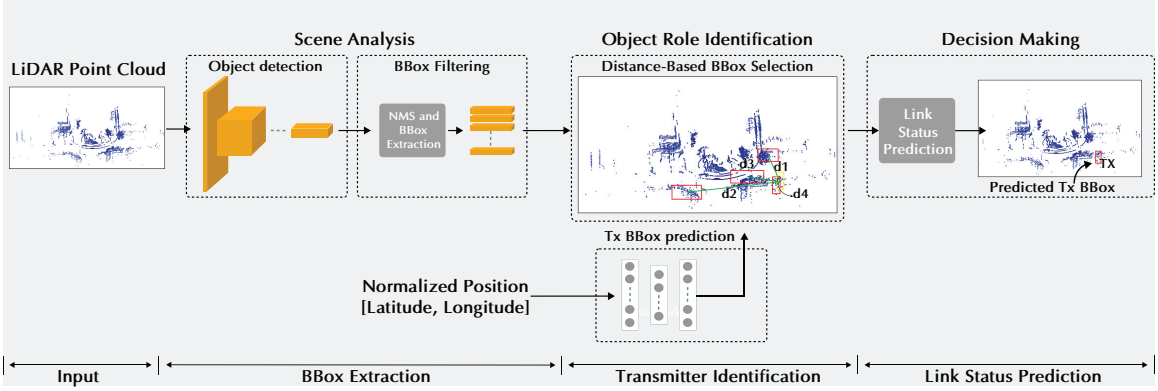


Figure 3.2: The Figure Presents the Proposed LiDAR-aided Link Status Identification Model That Leverages Both LiDAR and Position Data to Predict Whether the Link Status Is LOS or Not. It Highlights the Three Proposed Steps: (i) Scene Analysis, (ii) Object Role Identification, and (iii) Decision Making.

box centers of the transmitting candidate. The last step involves predicting whether the user is LOS or not. An in-depth overview of the three-step DNN architecture is provided below. The detailed solution for single data sample-based transmitter identification is presented in Fig. 3.2.

(i) Scene Analysis: In order to detect the transmitting candidate in real-wireless settings, the first step is to identify all the relevant objects in the scene (scene analysis). A pre-trained object detector is adopted for this purpose. The object detector generates bounding boxes of all objects in the scene that are of relevance to a wireless communication system. For example, in a scene depicting a city street, relevant objects include, but are not limited to, cars, trucks, buses, pedestrians, and cyclists. The generated set of bounding boxes are then filtered to only contain objects of the class that we are interested in, notably cars and buses. By using those output bounding boxes, the relevant-object matrix $\mathbf{B} \in \mathbb{R}^{N\times 3}$ is constructed such that each row has only the normalized coordinates of the center of a bounding box, where N is the number of relevant objects in the scene.

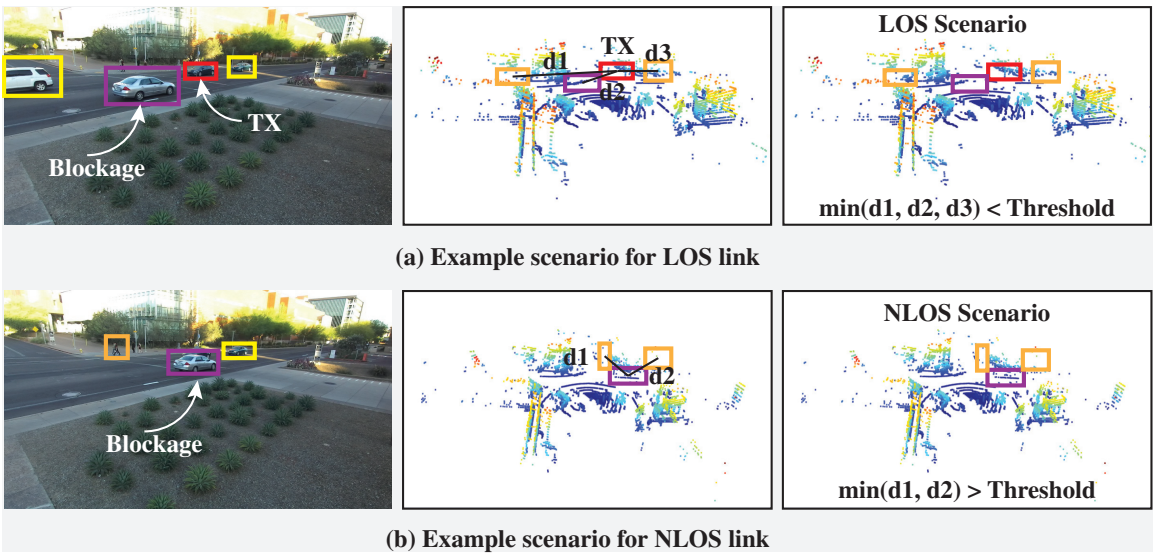


Figure 3.3: The Figure Presented Highlights the Decision Making Stage of the Proposed Solution. The Decision of Whether the User Is LOS or not Is Made by Computing a Threshold, i.e., If the Shortest Distance Is Greater than the Threshold, the User Is Blocked and Vice-versa.

(ii) Object Role Identification: In this step, both relevant-object matrix $\mathbf{B}$ and GPS position data are utilized to predict the bounding box center coordinates of the user. This step involves learning a prediction function that estimates the bounding box center of the user using the position data. The primary objective is to encode the relation between the position data and object location in the LiDAR point cloud data. The function is learned using a 4-layered feed-forward neural network. The prediction function f_Θ is parameterized by a set Θ representing the model parameters and learned from the dataset $\mathcal{D}$ of the labeled data samples. Since $\hat{\mathbf{b}}_{\text{Tx}}$ is an initial estimate that solely relies on position data, it is not expected to be a final prediction but merely an estimate. The idea here is to use the initial estimate in conjunction with the relevant-object matrix $\mathbf{B}$ to identify (or select) the object responsible for the radio signal (transmitter).

Table 3.1: Parameters of the 4-layered Neural Network

Layer	Parameters
Input Layer	Fully connected,size 2x20
Hidden Layer 1	Fully Connected,size 20x100
Activation Layer	ReLU Activation
Hidden Layer 2	Fully Connected,size 100x400
Output Layer	Fully Connected,size 400x3

In this step, using the additional modality, i.e., the position data, we can estimate the approximate center coordinates of the transmitter in the scene. We can then utilize these two pieces of information to identify the transmitter in the scene accurately. This is done using the nearest neighbor algorithm with a Euclidean distance metric. We first calculate the Euclidean distance between the predicted center coordinates and all the objects in $\mathbf{B}$. The element of $\mathbf{B}$ with the shortest distance to $\hat{\mathbf{b}}_{\mathrm{Tx}}$ is picked as the nearest neighbor and, hence, the predicted transmitter object. The assumption here is that a well-trained prediction function f_{Θ} can predict the center coordinates close to the actual values, and hence the Euclidean distance-based metric can help in accurately detecting the transmitter.

(iii) Decision Making: The final step involves deciding whether the user is LOS or not. As presented in the previous step, we first calculate the Euclidean distance between the predicted bounding box centers and the different objects in the scene. The next step involves selecting the object with the shortest distance from the predicted center. However, if this distance is greater than a particular value, it might represent a blocked user. In Fig. 3.3, we present the details of this stage. This decision of

whether the user is LOS or not is made by computing a threshold, i.e., if the shortest distance is greater than the threshold, the user is blocked and vice-versa. To compute the threshold, we utilize the training dataset and then evaluate the performance on the test set.

3.5 Development Dataset

To evaluate the performance of the proposed solution, we adopt several scenarios of the DeepSense6G dataset. Specifically, scenarios 31, 32, 33, and 34, consisting of multi-modal sensing and communications data, were utilized for this problem. A closer examination of the three-dimensional LiDAR data reveals that the data is too sparse to provide accurate beam prediction results. Small trees or shrubs block the laser at larger distances, making it challenging to view the transmitter. However, the data is still dense enough to visualize whether an object is being blocked or not. Sequences of data containing many multi-user blockage instances were identified from Scenarios 31, 32, 33, and 34 of the DeepSense6G dataset. The sequences were

Table 3.2: Details Of the Development Dataset

Scenario(s)	LOS Samples	NLOS Samples
Scenario 31	72	3
Scenarios 32, 33, 34	607	79

manually annotated using MATLAB's groundTruthLabeler, for two class labels; Tx and Distractor. Each bounding box represents a cuboid that contains the object of interest within the point cloud frame of reference. The bounding box is a 1×9 vector of the form [xctr, yctr, zctr, xlen, ylen, zlen, xrot, yrot, zrot], where:

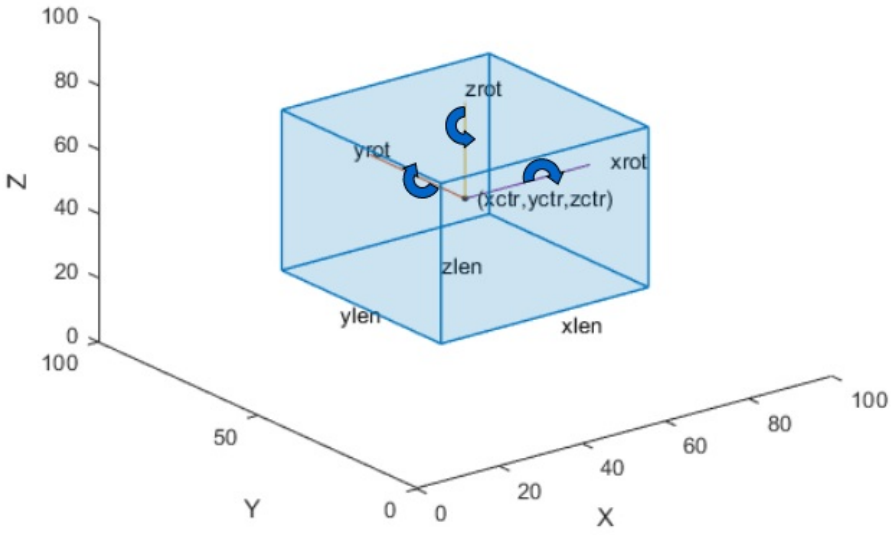


Figure 3.4: The Figure Presents How The 1×9 Vector Translate to a Cuboidal Bounding Box

- xctr, yctr, and zctr specify the center of the cuboid.
- xlen, ylen, and zlen specify the length of the cuboid along the x-, y-, and z-axis, respectively, before rotation has been applied.
- xrot, yrot, and zrot specify the rotation angles for the cuboid along the x-, y-, and z-axis, respectively. These angles are clockwise-positive when looking in the forward direction of their corresponding axes.

Fig. 3.4 presents how the above values translate to a cuboidal bounding box. The final development dataset consists of 422 training samples and 257 test samples. The test dataset contains all the samples from Scenario 31, meaning that the trained network would be tested on a data set containing samples from the previously unseen Scenario 31 along with samples from Scenario 32, 33, and 34.

Table 3.3: User Identification Performance

Parameter Observed	Value
Prediction MSE	7.8×10^{-6}
Tx ID Verification(Prediction Within 60cm)	91.4%

3.6 Performance Evaluation

In this section, we first discuss the neural network training parameters and the adopted evaluation metrics. Next, we present the numerical evaluation of the proposed solution.

Experimental Setup: The model is trained with an ADAM optimizer with an initial learning rate of 1×10^{-3}, for 150 epochs and a batch size of 128. All the simulations were performed on a single NVIDIA Quadro 6000 GPU using the PyTorch deep learning framework. We adopt the Mean Squared Error (MSE) loss to train the model. We utilize the top-1 accuracy metric as the primary method of evaluating the proposed solution.

Can 3D LiDAR and position data be utilized to identify the user?

To answer this question, we evaluate the proposed solution on the development dataset as described in Section 3.5. In Table 3.3, we present: (i) The MSE between the predicted and ground-truth centers of the user and (ii) the user identification accuracy on the test set. It is observed that the proposed solution achieved an MSE of 7.8×10^{-6} and an user identification of 91.4%. The extremely low MSE and high prediction accuracy of the proposed approach highlights that sensing-aided solutions can enable user identification in a multi-candidate scenario.

Can 3D LiDAR and position data aid in predicting link status? In Figure 3.5,

we present the probability distribution of the shortest distance between the predicted bounding box center and the nearest object for the LOS and NLOS cases. It also shows the selected threshold to distinguish between the LOS and NLOS cases. This plot is generated on the evaluation set of the development dataset which consists of data from the previous scenarios along with samples from the unseen scenario 31. The problem now reduces to a binary hypothesis problem[30] defined on the Euclidean distance x. The two distributions are the conditional distributions or more appropriately: the conditional densities for the two hypotheses, are mathematically defined as follows,

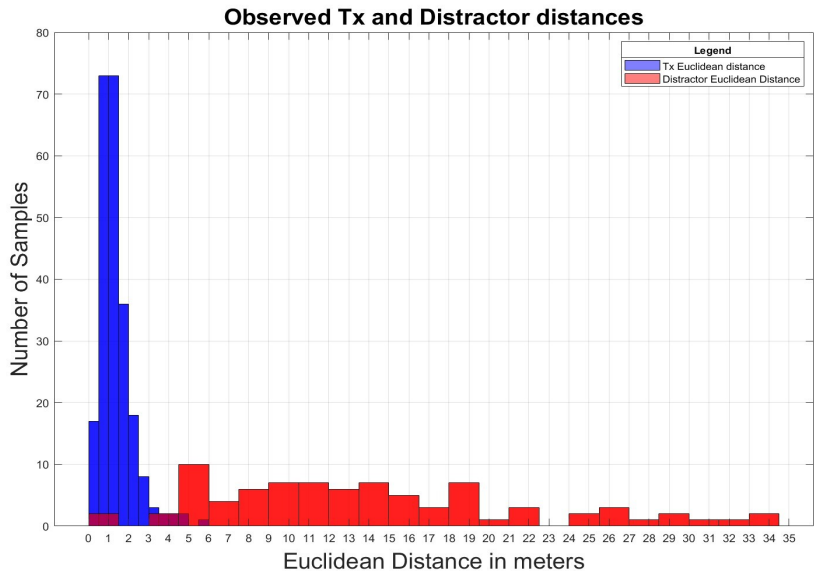
$$H_0 : X \in PDF(Line\,of\,Sight\,link) \tag{3.7}$$

$$H_1 : X \in PDF(Non\,Line\,of\,Sight\,link) \tag{3.8}$$

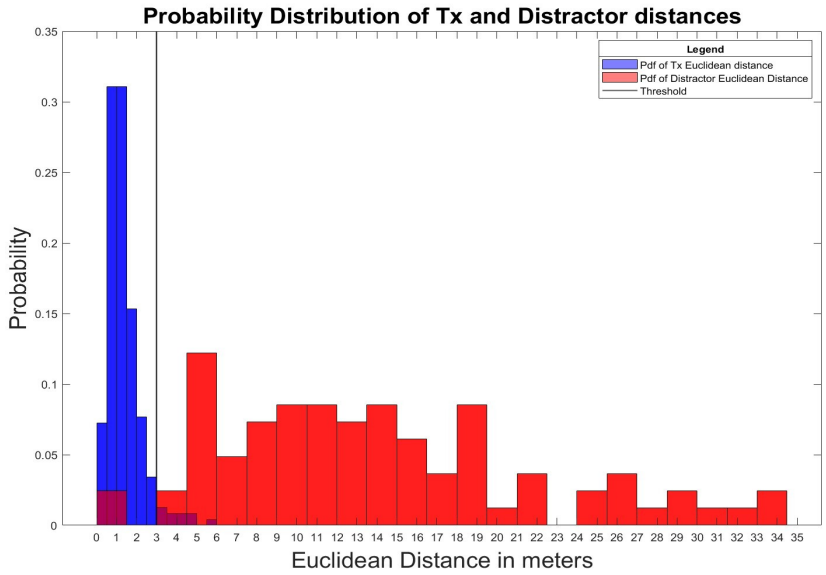
Computing the threshold may be performed using the Neyman-Pearson Likelihood ratio test which partitions the observation space into two regions Ω_{H_0} and Ω_{H_1} [30]. Partioning the observation space in this manner is constrained on the probability of false alarm. In the observed problem a false alarm occurring would be defined as *'Predicting that a blockage has occurred when a blockage has not occurred'*. The Neyman-Pearson Likelihood ratio test may be mathematically described as follows,

$$t_{LRT} \triangleq \frac{\mathbb{P}(x \mid H_1)}{\mathbb{P}(x \mid H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \eta \tag{3.9}$$

where t_{LRT} is the test statistic under the binary hypothesis , and η is the threshold. Upon evaluating the formulated LRT we obtain a value of $\eta = 3m$. We next compute the link status of the samples in the test dataset based on this calculated threshold. It is observed that for this small development dataset, we achieve a 87% link status prediction accuracy, highlighting the efficacy of the proposed solution.



(a)



(b)

Figure 3.5: The Figure (a) Presents the Distributions of the Shortest Distance Between the Predicted Bounding Box Center and the Nearest Object for The LOS and NLOS Cases; Figure (b) Presents the Probability Densities of the Two Distributions Presented in Figure (a)

Chapter 4

Conclusion

This thesis focused on developing a solution that addresses a key challenge in mmWave communications: the sensitivity of the mmWave link to blockages in the environment. First, we propose applying sensory information about the environment as a practical means by which one could obtain the channel impulse response. We then introduce the multi-modal sensing and communications dataset DeepSense 6G, which consists of co-existing wireless and sensory information. We discuss the implementation of the testbed utilized for collecting the dataset, including but not limited to the various sensory modalities employed, the procedure for collecting the data, post-processing the data, time synchronization, etc. Furthermore, we discuss the selection of 3D LiDAR as the sensory modality of choice and its advantages over images. Secondly, we propose a DNN-based bi-modal solution to identify a transmitter from a set of probable transmitting candidates by utilizing the 3D LiDAR point cloud and the GPS position of the transmitter. The performance of the transmitter identification is then evaluated. Consequently, we predict the bounding boxes of the transmitter based on the position and utilize the bounding boxes to generate a set of probability distribution functions based on Euclidean distances for both the LOS and NLOS cases. Finally, we propose a binary hypothesis test on the Euclidean distance distributions examined under the Neyman-Pearson Likelihood Ratio test and compute a threshold to predict the link status with high accuracy.

REFERENCES

[1] T. S. Rappaport, Y. Xing, O. Kanhere, S. Ju, A. Madanayake, S. Mandal, A. Alkhateeb, and G. C. Trichopoulos. Wireless communications and applications above 100 GHz: Opportunities and challenges for 6G and beyond. *IEEE Access*, 7:78729–78757, 2019. doi: 10.1109/ACCESS.2019.2921522.

[2] Jeffrey G. Andrews, Tianyang Bai, Mandar N. Kulkarni, Ahmed Alkhateeb, Abhishek K. Gupta, and Robert W. Heath. Modeling and analyzing millimeter wave cellular systems. *IEEE Transactions on Communications*, 65(1):403–430, 2017. doi: 10.1109/TCOMM.2016.2618794.

[3] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[4] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[5] Jorge Beltrán, Carlos Guindel, Francisco Miguel Moreno, Daniel Cruzado, Fernando García, and Arturo De La Escalera. Birdnet: A 3d object detection framework from lidar information. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3517–3523, 2018. doi: 10.1109/ITSC.2018.8569311.

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[7] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[8] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11784–11793, June 2021.

[9] Gouranga Charan, Muhammad Alrabeiah, and Ahmed Alkhateeb. Vision-aided 6G wireless communications: Blockage prediction and proactive handoff. *IEEE Transactions on Vehicular Technology*, 70(10):10193–10208, 2021. doi: 10.1109/TVT.2021.3104219.

[10] João Morais, Arash Behboodi, Hamed Pezeshki, and Ahmed Alkhateeb. Position aided beam prediction in the real world: How useful gps locations actually are?, 2022.

[11] Shuaifeng Jiang, Gouranga Charan, and Ahmed Alkhateeb. Lidar aided future beam prediction in real-world millimeter wave v2i communications, 2022.

[12] Shunyao Wu, Muhammad Alrabeiah, Chaitali Chakrabarti, and Ahmed Alkhateeb. Blockage prediction using wireless signatures: Deep learning enables real-world demonstration. *arXiv preprint, arXiv:2111.08242*, 2021.

[13] A. Alkhateeb, G. Charan, T. Osman, A. Hredzak, and N. Srinivas. DeepSense 6G: A large-scale real-world multi-modal sensing and communication dataset. *available on arXiv*, 2022. URL `https://www.DeepSense6G.net`.

[14] Robert W Heath, Nuria Gonzalez-Prelcic, Sundeep Rangan, Wonil Roh, and Akbar M Sayeed. An overview of signal processing techniques for millimeter wave mimo systems. *IEEE journal of selected topics in signal processing*, 10(3): 436–453, 2016.

[15] M. Bennis, M. Debbah, and H. V. Poor. Ultrareliable and low-latency wireless communication: Tail, risk, and scale. *Proceedings of the IEEE*, 106(10):1834–1853, 2018.

[16] Jeffrey G Andrews, Stefano Buzzi, Wan Choi, Stephen V Hanly, Angel Lozano, Anthony CK Soong, and Jianzhong Charlie Zhang. What will 5G be? *IEEE Journal on selected areas in communications*, 32(6):1065–1082, 2014.

[17] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi. Improved handover through dual connectivity in 5g mmwave mobile networks. *IEEE Journal on Selected Areas in Communications*, 35(9):2069–2084, 2017. doi: 10.1109/JSAC.2017.2720338.

[18] M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi. Multi-connectivity in 5G mmWave cellular networks. In *2016 Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pages 1–7, 2016.

[19] D. Aziz, J. Gebert, A. Ambrosy, H. Bakker, and H. Halbauer. Architecture approaches for 5G millimetre wave access assisted by 5G low-band using multi-connectivity. In *2016 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6, 2016. doi: 10.1109/GLOCOMW.2016.7848840.

[20] N. H. Mahmood and H. Alves. Dynamic multi-connectivity activation for ultra-reliable and low-latency communication. In *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*, pages 112–116, 2019. doi: 10.1109/ISWCS.2019.8877325.

[21] A. Alkhateeb, I. Beltagy, and S. Alex. Machine learning for reliable mmwave systems: Blockage prediction and proactive handoff. In *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 1055–1059, Nov 2018.

[22] Muhammad Alrabeiah, Andrew Hredzak, and Ahmed Alkhateeb. Millimeter wave base stations with cameras: Vision-aided beam and blockage prediction. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pages 1–5, 2020. doi: 10.1109/VTC2020-Spring48590.2020.9129369.

[23] Muhammad Alrabeiah and Ahmed Alkhateeb. Deep learning for mmwave beam and blockage prediction using sub-6 GHz channels. *IEEE Transactions on Communications*, 68(9):5504–5518, 2020. doi: 10.1109/TCOMM.2020.3003670.

[24] Umut Demirhan and Ahmed Alkhateeb. Radar aided proactive blockage prediction in real-world millimeter wave systems. In *Proc. of IEEE ICC, arXiv preprint, arXiv:2111.14805*, 2021.

[25] Umut Demirhan and Ahmed Alkhateeb. Integrated sensing and communication for 6g: Ten key machine learning roles, 2022. URL `https://arxiv.org/abs/2208.02157`.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[27] Gouranga Charan and Ahmed Alkhateeb. Computer vision aided blockage prediction in real-world millimeter wave deployments, 2022.

[28] A. Alkhateeb, G. Leus, and R. W. Heath. Limited feedback hybrid precoding for multi-user millimeter wave systems. *IEEE Transactions on Wireless Communications*, 14(11):6481–6494, 2015.

[29] Muhammad Alrabeiah and Ahmed Alkhateeb. Deep learning for TDD and FDD massive mimo: Mapping channels in space and frequency. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 1465–1470, 2019. doi: 10.1109/IEEECONF44664.2019.9048929.

[30] S.M. Kay. *Fundamentals of Statistical Signal Processing: Detection Theory*. Prentice-Hall Inc., 1998.