Low-Cost Experimental Platform for AI-Based Beam Management Using WiFi Radios

Valdinei Conceição, Flávio Nunes, João Borges, Cleverson Nahum, Ilan Correa, Silvia Lins, Aldebaro Klautau

Abstract— **Millimeter-wave (mmWave) beam management is critical for 5G/6G networks but faces experimental challenges due to costly hardware and lack of repeatable testing. This work introduces a novel low-cost platform using modified commercial-off-the-shelf (COTS) WiFi radios for beam management research. We propose a comprehensive solution featuring custom firmware for programmable codebooks and real-time metrics like Received Signal Strength Indicator (RSSI), combined with a robotic platform enabling automated mobility and synchronized multi-modal data collection. Experimental validation demonstrates the effectiveness of the platform in generating rich datasets, verifying codebook performance, and bridging the simulation-reality gap for AI-driven beam management systems.**

Keywords— **Beamforming, beam management, low-cost, millimeter-wave, robotic, synchronization.**

I. Introduction

High-frequency directional communication is pivotal for 5G and emerging 6G networks, offering unprecedented bandwidth to meet growing demands for low-latency, high-throughput applications. However, these systems face inherent challenges, including severe signal attenuation, sensitivity to blockages, and the complexity of maintaining precise beam alignment in dynamic environments [1]. Despite the critical role of beam management in enabling reliable directional links in these high-frequency wireless systems, experimental validation of novel strategies, particularly those involving artificial intelligence (AI) driven beam alignment, remains challenging. This is primarily due to high hardware cost barriers and limited access to programmable radios. Existing experimental platforms often prioritize either flexibility or affordability, but rarely both, creating a significant gap for reproducible, large-scale studies in dynamic environments.

Recent studies explored the use of AI for enhancing beam management in millimeter-wave (mmWave) systems. As noted by Sun et al. [2], AI-based beam prediction and model selection techniques, especially those tailored to specific propagation environments, can outperform traditional static approaches, reducing overhead and improving link reliability. The integration of multimodal data, such as visual information and wireless measurements, has shown significant promise in addressing these challenges. For instance, works like [3] demonstrated how Light Detection and Ranging (LiDAR) data can be effectively leveraged for line-of-sight (LoS) detection and to reduce the overhead associated with mmWave beam selection, by providing rich contextual awareness of the environment. Similarly, visual data streams can complement wireless metrics to predict optimal beam directions and adapt to dynamic changes in the channel. Nevertheless, the deployment of these solutions in real-world scenarios demands experimental platforms capable of collecting synchronized, multimodal data under diverse conditions.

Several platforms have been proposed in the literature to support such studies. The M-Cube [4] platform, for instance, is a fully programmable mmWave platform that supports hybrid beamforming experiments. However, achieving its full programmability often comes at a high cost, typically involving specialized hardware and significant engineering effort in reverse-engineering and low-level radio control in proprietary devices. This leads to considerable complexity in deployment and operation, limiting accessibility for researchers.

Beyond these high cost, extensively customizable solutions, other efforts have explored more accessible hardware. Wang et al. [5] designed an experimental mmWave vehicle-to-infrastructure (V2I) platform using commercial 60 GHz WiFi radios to evaluate vehicle-to-everything (V2X) performance in mobility scenarios, their platform offers valuable insights into real-world deployment conditions but lacks automation and repeatability, relying heavily on manual configuration and rigid test setups. Similarly, the HawkRover [6] platform proposes robot-based beam training experiments utilizing multi-sensor fusion, but the temporal alignment of data from these diverse sensors is not straightforward due to varying data frequencies, leading to complexities in data processing. The work [7] also illustrates the feasibility of compressive sensing for beam alignment using off-the-shelf devices, yet their experiments were constrained by static configurations and required extensive post-processing. The limitations of manual data collection and hardware complexity highlight a gap in accessibility and automation of mmWave research platforms.

To address these gaps, three key requirements emerge: (1) compatibility with low-cost, widely available transceivers; (2) automated data collection for repeatability in mobility/blockage scenarios; and (3) tight synchronization between wireless metrics and environmental context (e.g., vision sensors). Meeting these criteria would accelerate research on adaptive beam management while bridging simulation to reality discrepancies.

This paper introduces a low-cost and flexible mmWave experimental platform designed to address these challenges. Our system is built upon modified 802.11ad radios, which

V. Conceição, F. Nunes, J. Borges, C. Nahum, I. Correa and A. Klautau are with LASSE - Telecommunications, Automation and Electronics Research and Development Center, Belém-PA, Brazil. S. Lins is with the Innovation Center, Ericsson Telecomunicações S.A, Brazil. E-mails: [valdinei.conceicao, flavio.nunes, joao.tavares.borges]@itec.ufpa.br, [cleversonahum, ilan, aldebaro]@ufpa.br, silvia.lins@ericsson.com.

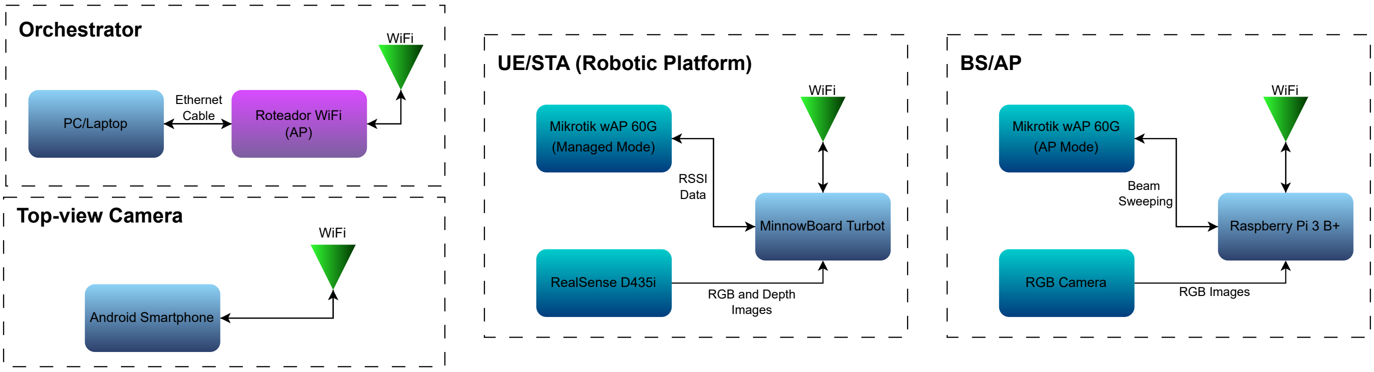

Fig. 1: General overview of the mmWave platform.

enable fine-grained control of beamforming and access to low-level radio metrics. It features a solid data collection framework based on a synchronized client-server architecture, capable of gathering data streams including Received Signal Strength (RSS), color (RGB) images, and depth information. A custom-built robot developed in-house is also integrated to enable automated measurements along predefined paths. This infrastructure gives researchers the ability to conduct indoor experiments and generate multimodal datasets for the development and validation of mmWave communication strategies, machine learning models, and beam management algorithms under diverse and realistic conditions.

The primary contribution of this work is the design and validation of the experimental platform itself. While it successfully collects synchronized wireless and visual data, the development of specific AI models that process this information for real-time beam management is the intended next step and is considered future work.

II. Proposed Low-Cost mmWave Experimental Platform

In this section, we explore our proposed mmWave experimental platform designed to facilitate wireless experimentation in both static and mobile scenarios. Our architecture relies on commercial-off-the-shelf (COTS) mmWave radios, which we have modified with custom firmware to enable access to low-level metrics and support custom beamforming. A computer automates the data collection process and controls the radios remotely. Additionally, our setup supports experiments with mobility by integrating a programmable robot platform capable of walking in a predefined line path.

Before diving into implementation details, we provide a general overview of the system in Fig. 1. The platform's main components include one radio configured as an Access Point (AP) and another operating in managed mode. To coordinate their behavior, we introduce an external computer (Orchestrator), which is responsible for configuring the radios, initiating transmissions, and storing the collected data.

To extract data from the mmWave radios without requiring hardware modifications, we selected the radio model wAP 60G from MikroTik, due to its wide availability, lower costs, and strong similarities with the Airfide Sparrow+, previously used in some works [5] [8]. The wAP 60G operates according to the IEEE 802.11ad standard [9]. The radio is equipped with a 6×6 Uniform Planar Array (UPA) and integrates a Qualcomm baseband chipset that provides a reference clock of around 7.5 GHz, enabling it to switch among four channels (centered at carrier frequencies of 58.32, 60.48, 62.64, and 64.80 GHz).

The default operating system of the radio, RouterOS, provides very limited access to operational parameters, making it unsuitable for our experiments. To overcome this barrier, we replaced it with a customized version of OpenWRT [1], which offers greater control over the device. We also modified the firmware of the `wil6210` baseband module, compliant with the IEEE 802.11ad standard, to access key internal metrics such as Received Signal Strength Indicator (RSSI), angle of departure (AoD), angle of arrival (AoA), and time of flight (ToF), as well as to enable manual beam selection, that were features not available by default.

To enable access, we applied the same RSSI extraction techniques described in [5] [8], leveraging the fact that our radio employs the same phased array and wil6210 firmware. Following the approach in these works, the Nexmon framework [10] was utilized to modify the firmware of the mmWave module, redirecting the desired information to a memory section accessible by developers. Minor code and compilation adjustments were required to ensure compatibility with MikroTik's CPU architecture.

A Python server runs on the radio, connected to an external computer via the Ethernet port. The computer sends commands to read the internal metrics (e.g., RSSI) and to control functionalities such as manual beam selection, as previously described.

A. Data Collection Architecture

To extensively and automatically collect RSSI measurements, we developed a client-server architecture in C++, where one computer named Orchestrator manages N connected clients. Each client is equipped with hardware to capture information, such as mmWave radios or RGB and depth images from attached cameras.

The system is designed to be scalable, allowing the connection of an arbitrary number of clients by adding more radios, cameras, or other data acquisition components, depending on limiting factors such as the maximum capacity of the network channel between them. In our experiments, we utilized the following four devices:

- A laptop with an Intel i3-3227U processor and 2GB of RAM, acting as the Orchestrator.
- A Raspberry Pi 3 B+ connected to the radio operating in AP mode, with a 1080p RGB camera, used as a client to collect images from the radio's perspective.
- A MinnowBoard Turbot connected to the radio in Station (STA) mode, equipped with an Intel RealSense D435i camera, responsible for collecting RSSI data along with RGB and depth images.
- An Android smartphone, responsible for capturing wide-angle top-view images of the measurement scenario.

B. Synchronization Mechanism

To minimize capture delay between clients and ensure synchronized measurements across devices, we implemented a centralized request-based communication model. The Orchestrator sends a capture request command to all connected clients simultaneously. Upon receiving the command, each client captures data from its assigned sensors and immediately transmits back to the Orchestrator.

We empirically determined that a 200 ms interval between requests provides the best trade-off between data throughput and reliability. This timing accounts for internal delays in devices, such as the RSSI update rate of the mmWave radio's memory, which can lead to inconsistent or duplicated values at shorter intervals.

To ensure synchronization, the Orchestrator also functions as a local network time protocol (NTP) server, allowing all clients to synchronize their system clocks with it, making a minimal delay between the clocks. During each capture cycle, the client records a local timestamp at the moment of data acquisition, while the Orchestrator records the reception timestamp. These timestamps are stored with the captured data for post-processing, enabling precise temporal alignment and verification of the synchronization accuracy. This precise timestamping is crucial for feeding AI systems, ensuring that inputs from different sensors refer to the same (or closest possible) instant in time, which is vital for accurate model training and inference.

C. Communication Channel

An isolated local network was configured for the system. The Orchestrator connects to this network via Ethernet, while the clients are connected over a 2.4 GHz Wi-Fi interface. This provides a balance between mobility and reliability for the devices. Communication between the Orchestrator and the clients is implemented over TCP sockets, each client maintains a persistent connection to the Orchestrator, listening for capture commands.

The client response packet is designed to be as compact as possible to reduce transmission latency and maximize data throughput. The packets begin with a header containing three essential fields: the device type (used to identify the device role, such as camera, radio in STA mode, or radio in AP mode), the total size of the payload, and the timestamp of data capture. This is followed by the payload itself, containing compressed image data or RSSI values.

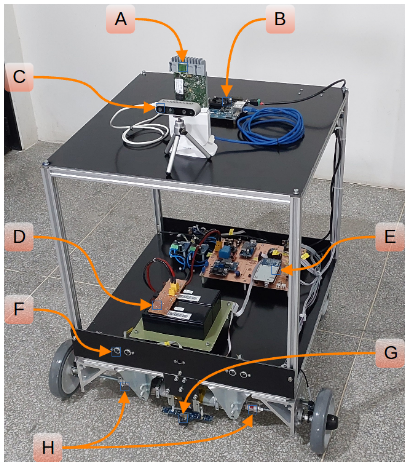

Fig. 2: Robotic platform. A) mmWave radio. B) Single-board computer. C) RGB camera with depth sensor. D) Battery pack. E) Robot controller board. F) Distance sensors. G) Follow line sensor. H) Electric motors.

D. Robotic Platform

To perform automated movement in mmWave signal measurements with precision, we designed a robotic platform named Acquisition Control Systems Automatic Implementation (ACAI). This platform was developed in-house and mechanically structured to carry all the necessary components for data collection, including the radio and a single-board computer. It provides power to all onboard equipment, as illustrated in Fig. 2.

The ACAI embedded system integrates several sensor types, including voltage and current sensors for monitoring battery performance, ultrasonic sensors for obstacle avoidance, and a line-following sensor for path tracking. The robot is controlled by a microcontroller based on the 8-bit Atmega 2560 AVR core. Its firmware execute a line follower with digital proportional–integral–derivative (PID) controller algorithm for motor speed control [11].

For mobility, a line-tracking sensor was used to detect a path drawn on the floor, it was necessary to create a high-contrast black line on a white background for the best performance of the sensor and the line following algorithm [12] [13].

III. Results

Having described the proposed mmWave experimental platform, this section now presents key results demonstrating its capabilities.

A. Codebook Customization and Visualization

Using the modified operating system and firmware, it is possible to read and write the desired amplitude and phase values for each antenna element of the 6×6 UPA, loading this information during device initialization to define its codebook. Furthermore, following the approach described in [8], specific

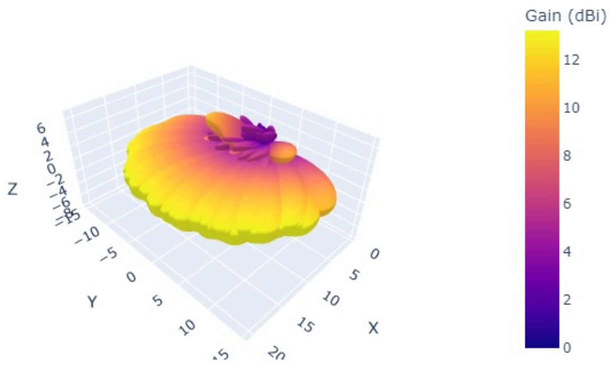

Fig. 3: Azimuth sweeping codebook irradiation pattern.

beam values can be updated almost in real time without requiring a reboot. This allows for beam-steering implementations in which the codebook is initialized with a single codeword that can be dynamically reconfigured to point in a desired direction. This contrasts with the default behavior of the device, where it performs beam sweeping across a predefined set of beams to find the optimal direction.

An example of a custom codebook created to demonstrate the flexibility of beam steering, validate our phase rotation implementation, and visualize the applied modifications is one that performs a sweep in the azimuth direction, covering a total variation of approximately $100°$. Fig. 3 shows the three-dimensional irradiation pattern of the 50 different beams defined in the codebook, where the color bar represents the gain in dBi.

B. Data Capture

Measurement campaigns using the developed system were conducted in a complex indoor environment with an area of approximately 6.5×6 meters, including multiple workbenches, chairs, and other obstacles for the beamforming. The measurement scenario is depicted in Fig. 4. In this setup, the transmitter (Tx) radio is represented by the red semi-circle at the top of the figure and is configured with the default MikroTik codebook, which includes 64 directional beams spanning approximately $60°$ in both azimuth and elevation.

The receiver (Rx) radio is mounted on top of the ACAI robot, which autonomously follows a predefined line path through the environment. For reception, the radio is configured with a default quasi-omnidirectional beam. As the robot completes one full lap along the path, data is captured at sixteen distinct positions. Some of these positions are illustrated in the figure as colored squares, highlighting the spatial distribution of the measurements across the test environment.

Fig. 5 illustrates the point-of-view (POV) of both radios at measurement position 31. The Rx radio collects both RGB and depth images but at this position does not have a direct view to the Tx. Conversely, the Tx radio captures only RGB images and maintains LoS with the Rx, which is highlighted by the orange box in the figure.

To analyze the signal strength, the collected RSSI values were converted to RSS using the linear model proposed in [14], given by

$$RSS = 0.0651 \cdot RSSI - 74.3875, \qquad (1)$$

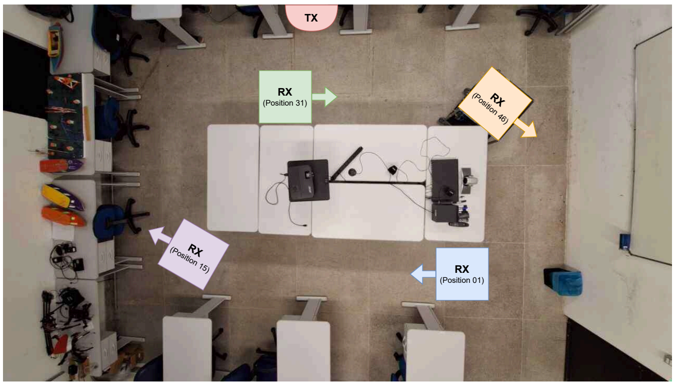

Fig. 4: Top-view illustration of measurement campaign.

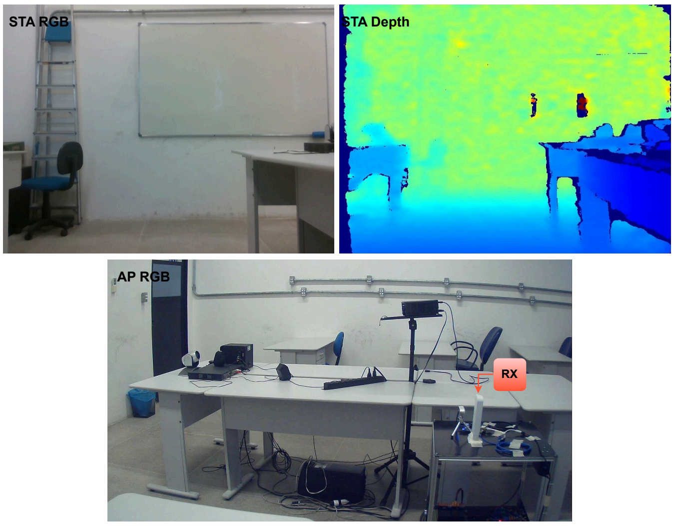

Fig. 5: Tx and Rx radios POV, from the 31st position.

the resulting values were validated using the `iw dev wlan0 scan` linux command, where the reported signal corresponds to the strongest RSS observed during the scan.

The per-beam RSS results for the four selected measurement positions are shown in Fig. 6. The graph is scaled in dBm, and the best-performing beam at each location is highlighted in bright red. Additionally, the figure includes the irradiation pattern of the selected best beam, visualized from the Tx radio POV, which is positioned at the origin $(0, 0, 0)$. By comparing Fig. 4 and Fig. 6, it is possible to observe that the main lobe direction of the best beam of the Tx radio aligns well with the horizontal position of the Rx radio, validating the spatial consistency of the captured data.

IV. Conclusions

This work presented a low-cost and flexible mmWave platform built with modified COTS radios, capable of collecting synchronized multimodal data to support research in beam management. The platform integrates a custom firmware stack, a client-server architecture for multi-device synchronization, and an autonomous robotic platform to automate measurements in indoor environments. Despite relying only on RSSI values for beamforming analysis, the system proved effective in generating meaningful datasets, validating measurement

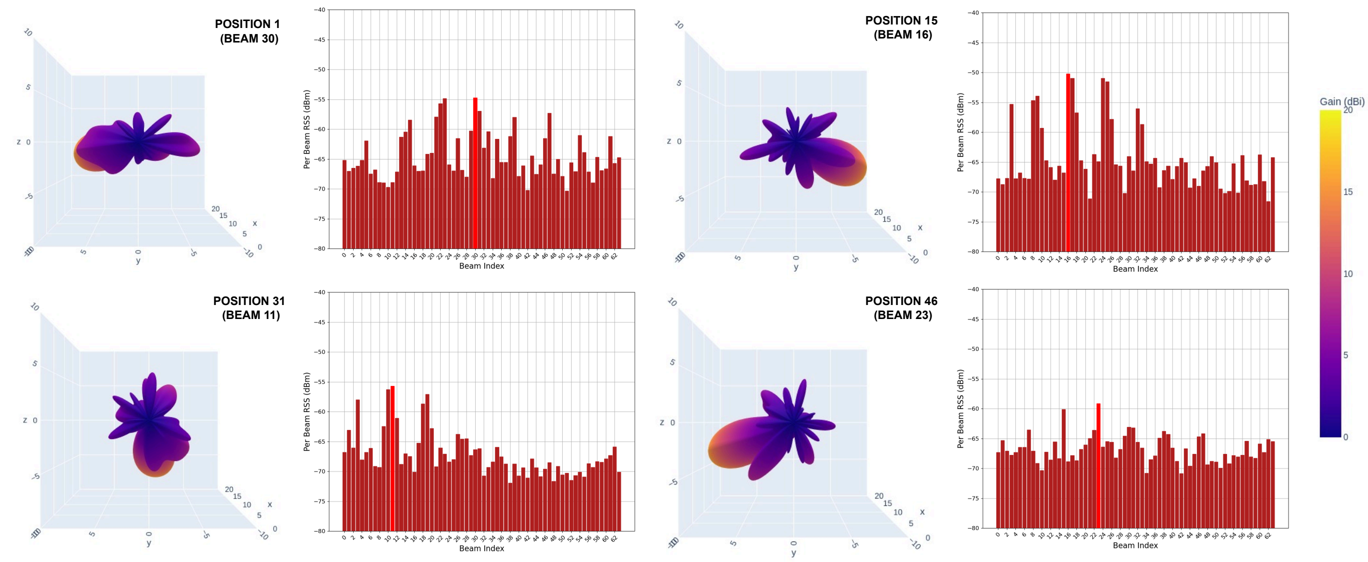

Fig. 6: Per-beam RSS results with the respective best beam irradiation pattern.

quality, and enabling codebook visualization for directional beam experiments.

The data acquisition pipeline, including the robot, camera modules, and radios, was successfully validated in controlled measurement campaigns. The resulting dataset demonstrates the platform's potential for use in AI-based beam management and other learning-driven applications. The most direct avenue for future work is to leverage the collected data to design and train models that use visual inputs to predict optimal beam configurations. Future work can also integrate the data capture process more tightly with ACAI's movement, improving the temporal resolution of the collected measurements. Additionally, while the platform reduces cost and complexity, it currently relies on RSSI metrics, limiting granularity compared to full-channel state information. Future work could integrate advanced metrics (e.g., channel impulse response) and expand to multi-node scenarios.

Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) – Finance Code 001; the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq); the Brasil 6G project (01245.020548/2021-07), supported by RNP and MCTI; the Innovation Center, Ericsson Telecomunicações S.A., Brazil; OpenRAN Brazil - Phase 2 project (MCTI grant Nº A01245.014203/2021-14); Project Smart 5G Core And MUltiRAn Integration SAMURAI (MCTIC/CGI.br/FAPESP under Grant 2020/05127-2).

References

[1] P. J. Mateo, *Network management and control for mmWave communications*. PhD thesis, Universidad Carlos III de Madrid, 2021.

[2] C. Sun, L. Zhao, T. Cui, H. Li, Y. Bai, S. Wu, and Q. Tong, "Ai model selection and monitoring for beam management in 5g-advanced," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 38–50, 2024.

[3] A. Klautau, N. González-Prelcic, and R. W. Heath Jr, "Lidar data for deep learning-based mmwave beam-selection," *IEEE Wireless Communications Letters*, vol. 8, pp. 909–912, June 2019.

[4] R. Zhao, T. Woodford, T. Wei, K. Qian, and X. Zhang, "M-cube: a millimeter-wave massive mimo software radio," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, MobiCom '20, (New York, NY, USA), Association for Computing Machinery, 2020.

[5] S. Wang, J. Huang, and X. Zhang, "Demystifying millimeter-wave v2x: towards robust and efficient directional connectivity under high mobility," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, MobiCom '20, (New York, NY, USA), Association for Computing Machinery, 2020.

[6] E. Zhu, H. Sun, and M. Ji, "Hawkrover: An autonomous mmwave vehicular communication testbed with multi-sensor fusion and deep learning," 2024.

[7] D. Steinmetzer, D. Wegemer, M. Schulz, J. Widmer, and M. Hollick, "Compressive millimeter-wave sector selection in off-the-shelf ieee 802.11ad devices," in *Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies*, CoNEXT '17, (New York, NY, USA), p. 414–425, Association for Computing Machinery, 2017.

[8] S. Wang, J. Huang, X. Zhang, H. Kim, and S. Dey, "X-array: approximating omnidirectional millimeter-wave coverage using an array of phased arrays," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, MobiCom '20, (New York, NY, USA), Association for Computing Machinery, 2020.

[9] R. S. I.-R. International Telecommunication Union, "IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 3: Enhancements for Very High Throughput in the 60 GHz Band," Standard IEEE Std 802.11ad-2012, 2012.

[10] M. Schulz, D. Wegemer, and M. Hollick, "Nexmon: The c-based firmware patching framework," 2017.

[11] V. Balaji, M. Balaji, M. Chandrasekaran, M. A. khan, and I. Elamvazuthi, "Optimization of pid control for high speed line tracking robots," *Procedia Computer Science*, vol. 76, pp. 147–154, 2015. 2015 IEEE International Symposium on Robotics and Intelligent Sensors (IEEE IRIS2015).

[12] M. Pakdaman, M. M. Sanaatiyan, and M. R. Ghahroudi, "A line follower robot from design to implementation: Technical issues and problems," in *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, vol. 1, pp. 5–9, Feb 2010.

[13] S. Shirmohammadi and F. Baghbani, "Design and implementation of a line follower robot," in *2024 10th International Conference on Artificial Intelligence and Robotics (QICAR)*, pp. 268–271, Feb 2024.

[14] Y. Song, C. Ge, L. Qiu, and Y. Zhang, "2ace: Spectral profile-driven multi-resolutional compressive sensing for mmwave channel estimation," in *Proceedings of the Twenty-Fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, MobiHoc '23, (New York, NY, USA), p. 41–50, Association for Computing Machinery, 2023.