

Radar odometry using mmWave technology for SLAM applications.

Leveraging mmWave Radar Sensor Technology for odometry estimation.

Luis Fernando Rodriguez Gutierrez
Fachhochschule Dortmund
M.Eng. Embedded Systems Engineering
luis.rodriguez001@stud.fh-dortmund.de

Abstract—Radar technologies have recently emerged as a promising alternative to traditional odometry methods, which often rely on cameras, LiDAR, wheel encoders, inertial sensors, or GPS. Unlike these modalities, millimeter-wave (mmWave) radar offers robustness under adverse conditions such as fog, rain, or low light, while directly providing both range and Doppler velocity measurements. Previous work has demonstrated that radar can support odometry through global iterative closest point (ICP) alignment of point clouds. Building on this foundation, this work focuses on ego-motion estimation using a dual mmWave radar system complemented by an inertial sensor. The dual-radar configuration is designed to increase spatial coverage and reduce ambiguity in radar detections, addressing a key limitation of single-radar odometry approaches.

Experimental evaluations were conducted in both enclosed environments (laboratory) and open environments (university parking lot). Results show that the proposed framework provides consistent and reliable ego-motion estimation, highlighting the potential of mmWave radar as a cost-efficient substitute or complement to conventional odometry sources in autonomous navigation.

Index Terms—Radar Odometry, mmWave Radar, Dual-Radar Configuration, Clustering, ICP, Doppler Velocity, Submap Aggregation, Ego-Motion Estimation

I. INTRODUCTION

Accurate and reliable ego-motion estimation is a fundamental requirement for mobile robotic systems and autonomous vehicle solutions. It is the basis for localization, mapping, and navigation, and errors in this stage directly affect the overall performance of autonomous systems.

Traditionally, odometry has been estimated using a combination of wheel encoders, inertial measurement units (IMUs), and GPS. In more recent years, cameras and LiDAR have been widely used because they provide dense information about the environment, enabling precise feature extraction and recognition. However, these vision-based and LiDAR-based methods have significant drawbacks. Cameras are highly sensitive to illumination changes. LiDAR systems are costly, and their performance can degrade in adverse weather conditions such as fog, rain, or snow. Both methods also require high computational and memory resources. These limitations create the need for complementary sensing solutions that remain reliable under real-world conditions.

Millimeter-wave (mmWave) radar has emerged as a strong candidate to address these issues. Radar is compact, cost-efficient, and inherently robust to poor lighting and weather. A key advantage of radar is its ability to directly measure Doppler velocity. Unlike cameras or LiDAR, which require frame-to-frame comparisons to estimate motion, radar provides direct measurements of radial velocity. This not only indicates how fast the vehicle is approaching or moving away from an object, but also makes it possible to separate static structures from moving objects and to detect relative motion trends. These Doppler-based measurements provide additional constraints for ego-motion estimation, making radar a unique and valuable sensing modality.

Despite these advantages, radar data also presents challenges. The resulting point clouds are sparse and noisy, and they often include significant amounts of clutter. Radar also has lower angular resolution compared to LiDAR or cameras. These limitations make it difficult to directly apply traditional scan-matching techniques, which are usually designed for dense LiDAR point clouds. Previous work has shown that radar-only odometry and multimodal fusion can improve robustness, but challenges remain when dealing with sparsity, clutter, and the stability of scan registration.

This work investigates the use of mmWave radar sensors mounted on a Ninebot Go-Kart test platform. The system also integrates an IMU and an embedded processing unit. A visual overview of the setup, including sensor placement, is shown in Figure 1. This figure highlights the main components without going into detailed descriptions, which are provided later in the hardware description section.



Fig. 1: Ninebot test-vehicle system.

The contributions of this work can be summarized as follows:

- 1) A radar ego-motion pipeline using mmWave sensors and an IMU for rotation, minimizing hardware cost and system complexity.
- 2) Integration of Doppler velocity and RANSAC filtering to improve the separation of static and dynamic objects.
- 3) Submap aggregation to mitigate point cloud sparsity and improve alignment stability.
- 4) Object tracking via clusters to identify and filter dynamic objects from the ego-motion estimation.
- 5) Experimental validation using real-world data collected from a vehicle-mounted mmWave radar sensor.

II. SYSTEM DESIGN AND METHODOLOGY

A. Research Design

The main objective of this work is the development of a radar-based odometry system that estimates the vehicle's ego-motion using mmWave radar sensors mounted at the front of the vehicle, in combination with an IMU for rotation compensation. The motivation behind this approach is to explore radar as a cost-effective and robust alternative to vision- or LiDAR-based odometry, particularly in conditions where these methods tend to fail. This builds on prior evidence that radar can support instantaneous ego-motion estimation through Doppler velocity cues [12].

The system processes radar point cloud data enriched with range, angle, and Doppler velocity to extract accurate motion estimates for ego-velocity. This enables reconstruction of the vehicle's trajectory and provides valuable input for SLAM applications.

As shown in Figure 2, two mmWave radar sensors are mounted at the front of the vehicle, with overlapping fields of view to improve coverage and reduce ambiguity. This configuration represents the initial validation scenario, chosen deliberately as a simple and controlled setup to test the basic functionality of the dual-radar concept. Starting from this baseline allowed us to validate the pipeline step by step before extending the experiments to more complex environments such as open outdoor areas.

The dual-radar arrangement increases point density and stability compared to single-radar approaches, aligning with multimodal strategies for robust state estimation [13], [14].

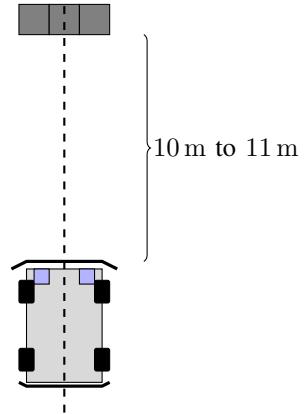


Fig. 2: Test scenario with dual front-mounted mmWave radar sensors. This setup served as the baseline validation environment before moving to more complex scenarios.

By fusing radial speed measurements obtained from Doppler with rotational information from the IMU, the proposed system is designed to remain resilient even in scenarios where LiDAR- or camera-based odometry would degrade. Each radar stream is first processed independently and then merged into a combined representation before feeding into the odometry pipeline.

1) Sub-Tasks: The research objective, together with the constraints of using a dual-radar sensor, implied several practical sub-tasks:

- Designing a modular pipeline to acquire and decode synchronized radar data from both sensors.
- Investigating suitable sensor configurations to balance field of view, chirp bandwidth, update rate, and detection density.
- Developing suitable mechanical mounts and selecting optimal sensor placement to ensure stability and maximize coverage.
- Applying RANSAC filtering on Doppler velocities to reject dynamic points and outliers.
- Implementing clustering methods to structure radar detections and isolate relevant features.
- Optimizing the raw radar point cloud using additional information provided by the sensor itself (e.g., SNR, RCS, or range validity) to improve reliability before odometry processing.
- Integrating Doppler velocity information into the odometry estimation process.
- Employing submap aggregation to mitigate sparsity and improve stability.
- Performing ICP alignment between submaps aggregated from both sensors to mitigate sparsity and noise.
- Evaluating the influence of the dual-sensor arrangement on odometry accuracy and robustness.
- Validating the complete system on real-world driving scenarios.

As each sub-task builds upon the results of the previous

one, the work followed an iterative and modular development approach, enabling gradual integration and continuous evaluation of the proposed system. The following sections detail the hardware design, pipeline methodology, and experimental validation.

B. System Hardware

The hardware setup combines mmWave radars, an inertial sensor, and auxiliary components into a compact test platform. In this section, we describe the role of each element, how they were mechanically integrated, and how the radar configuration was selected for odometry tasks.

1) Sensors: The sensing suite consists of:

a) mmWave Radar (IWR6843AOP):

The IWR6843AOPEVM development board from Texas Instruments features the IWR6843AOP, a high performance 4D mmWave FMCW sensor with Antenna On Package (AOP) design. Although IWR6843AOP is intended for industrial applications and its complementary chip, AWR6843AOP, for automotive applications, IWR6843AOP was used in this project because it is available in the form of this development board and the two chips are identical in terms of their functionalities, only differing in compliance with automotive industry [2]. Its small physical size, due to its AOP design, makes it an optimal choice for the desired mounting position, the go-kart's steering column.

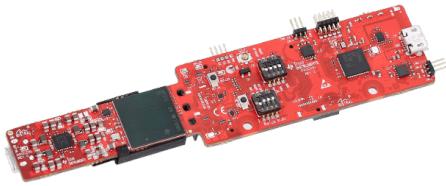


Fig. 3: IWR6843AOP sensor

Source: Texas Instruments, available at https://www.ti.com/ds_dgm/images/fbd_swrs219f.gif

The IWR6843AOP radar sensor operates within the frequency range of 60 GHz to 64 GHz and integrates 4 receive (RX) and 3 transmit (TX) antennas, radio frequency (RF) front-end stages, analog signal processing, and digital signal processing (DSP). It offers a wide range of communication interfaces including SPI, I2C, CAN-FD, UART and LVDS for raw data access and an Arm Cortex-R4F microcontroller for user-applications as shown in the Figure 4 [3].

As some operating parameters influence each other, their selection must be done carefully while observing the influence of the trade-offs involved. This could be referred to as "sensor tuning" and is a critical step because it directly impacts the system's accuracy and performance.

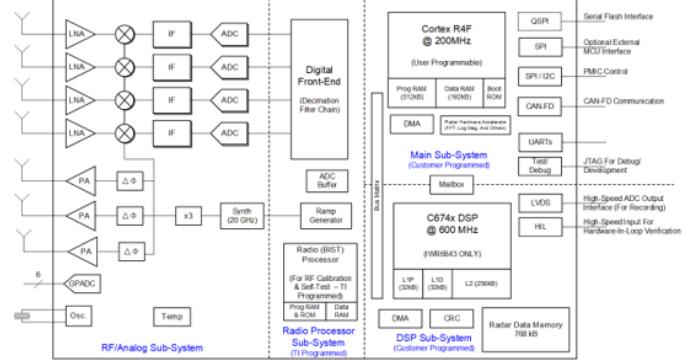


Fig. 4: IWR6843AOP internal block diagram.

Source: Texas Instruments, available at https://www.ti.com/ds_dgm/images/fbd_swrs219f.gif

The following operating parameters can be tuned:

- Frame rate
- Range resolution
- Maximum unambiguous range
- Maximum radial velocity
- Radial velocity resolution

Tuning these operating parameters introduces trade-offs by influencing each other in the following ways: The resulting

Tuning Parameter	Effect on Performance	Related HW Block	Trade-Off
Frame Rate	Higher FPS = faster updates but more processing load	C674x DSP, Radar Data Memory	Higher FPS reduces maximum range
Range Resolution	Higher resolution = better object separation	ADC, ID FFT (Range FFT)	Higher resolution reduces maximum range
Maximum Range	Determines farthest detectable object	RF Front-End, PA, LNA, ADC	Higher range lowers resolution
Radial Velocity Resolution	Improves speed accuracy	DSP, 2D FFT (Doppler FFT)	Higher resolution requires more chirps
Maximum Radial Velocity	Detects fast-moving objects	Chirp rate, TX Antennas, 2D FFT	Higher max velocity reduces resolution

TABLE I: Radar System Tuning Parameters and Trade-offs

overall accuracy of the velocity and distance measurements is again dependent on these operating parameters:

- **Radial velocity accuracy:** A fine balance between velocity resolution and frame rate must be maintained to ensure precise Doppler shift measurements. Lower resolution results in rounded velocity values, while an excessively high frame rate may introduce computational bottlenecks.
- **Distance accuracy:** Optimizing range resolution and maximum range ensures that detected objects are positioned accurately within the environment. Increasing range often sacrifices resolution, leading to potential inaccuracies in close-range detections.
- **Signal processing considerations:** The FFT calculation parameters directly affect both range and Doppler calculations, influencing the ability to distinguish between objects and detect small velocity variations.

This shows that finding exact values for the operation parameters by adjusting them while carefully watching their influences is crucial and heavily dependent on the particular application. The test scenario required a frame rate of approximately 30Hz to balance responsiveness and computational load, together with sufficient range and velocity coverage to capture typical vehicle dynamics.

The resulting configuration yielded the following operating parameters:

- Frame rate: 30 f s^{-1}
- Range resolution: 0.047 m
- Maximum unambiguous range: 9.78 m
- Maximum radial velocity: 8.01 m s^{-1}
- Radial velocity resolution: 0.51 m s^{-1}

Tuning and choosing the sensor's parameters carefully is extremely important as it defines the accuracy of therefore influences the reliability of the entire radar system. Fine-tuning these settings ensures that the sensor operates optimally.

b) Inertial Measurement Unit (IMU):

The MTi-G-710 from Xsens is a high-performance GNSS/INS module that integrates a 3D accelerometer, 3D gyroscope, 3D magnetometer, and barometer with an onboard GNSS receiver. Although designed as a complete navigation unit capable of delivering position and velocity estimates, in this project it was used specifically for its orientation outputs, provided in the form of quaternions, Euler angles, or rotation matrices [18]. Its compact form factor and proven sensor-fusion engine made it a suitable choice for integration into the test platform, complementing the radar subsystem by supplying real-time orientation references.

For the present setup, the device was configured to output quaternions, as this representation avoids singularities inherent in Euler angles and provides numerical stability for continuous tracking. These quaternions were used to track the vehicle attitude and to align radar point clouds within a consistent vehicle-centric coordinate frame. By relying on the MTi-G-710 internal fusion engine rather than implementing custom filtering, the system leveraged factory-calibrated algorithms for bias correction and drift reduction, minimizing processing overhead on the host platform.

To interface with the device and parse its MTData2 protocol, an open-source implementation from Scottapotamas was adopted [19]. This repository enabled robust packet parsing and quaternion extraction in Python, ensuring seamless integration of the IMU data stream into the radar processing pipeline.

c) Webcam:

A Logitech C270 HD webcam was included in the experimental platform for the sole purpose of video recording. Unlike the radar and IMU, the camera was not integrated into the sensing or processing pipeline and did not contribute any data to the odometry estimation system. Its role was limited to providing visual documentation of the test environment, allowing experiments to be reviewed and contextualized with synchronized video footage.

The Logitech C270 is a consumer-grade USB webcam capable of capturing video at a resolution of 1280×720 pixels and 30 Hz. Its compact design and ease of deployment made it a practical choice for recording without interfering with the radar sensors or mechanical mounts. The resulting video logs were used as qualitative references to better understand the test scenarios and to visually verify environmental conditions during system validation.



Fig. 5: MTi-G710 high-performance inertial navigation system (INS).

Source: Movella, available at <https://www.movella.com/sensor-modules/xsens-mti-7-gnss-ins>

2) Mechanical Integration: All sensors were mounted on a Ninebot Go-Kart test vehicle. Custom 3D-printed mounts were designed to ensure rigid placement and repeatability of radar orientation. The radars were positioned at the front of the vehicle with partially overlapping fields of view to improve coverage and reduce blind spots. The IMU was installed near the center of the kart to minimize rotational offsets. The webcam was mounted facing forward to log experiments for later analysis.

3) Radar Chirp Configuration: Radar performance is strongly determined by chirp parameters. For odometry, the configuration aimed to balance:

- **Range resolution** - to separate nearby objects.
- **Update rate** - to maintain real-time odometry.
- **Field of view** - to capture sufficient static structures.

This setup ensured compatibility with the subsequent pipeline stages, particularly Doppler filtering, clustering, and ICP alignment. While the exact numerical values are described later in the experimental section, the conceptual rationale here highlights how chirp design affects odometry robustness.

III. SUMMARY AND OUTLOOK

Insert conclusion here

REFERENCES

- [1] Segway Inc., “Ninebot Go-kart PRO product page”, 2025, Webpage. [Online]. Available: <https://de-de.segway.com/products/ninebot-gokart-pro>
- [2] Texas Instruments, “IWR1642: difference between AWR and IWR parts”, 2025, Webpage. [Online]. Available: <https://e2e.ti.com/support/sensors-group/sensors/f/sensors-forum/742730/iwr1642-difference-between-awr-and-iwr-parts>
- [3] Texas Instruments, “IWR6843AOPEVM product page”, 2025, Webpage. [Online]. Available: <https://www.ti.com/tool/IWR6843AOPEVM>
- [4] Texas Instruments, “User’s Guide mmWave Demo Visualizer,” 2020, Online Document. [Online]. Available: <https://www.ti.com/lit/ug/swru529c/swru529c.pdf?ts=1742817596204>.
- [5] Texas Instruments, “Understanding UART Data Output Format”, 2025, Webpage. [Online]. Available: https://dev.ti.com/tirex/content/radar_toolbox_2_20_00_05/docs/software_guides/Understanding_UART_Data_Output_Format.html
- [6] Texas Instruments, “mmWave Sensing Estimator”, 2025, Webpage. [Online]. Available: <https://dev.ti.com/gallery/view/mmwave/mmWaveSensingEstimator/ver/2.5.1/>
- [7] ub4raf, “Ninebot-PROTOCOL”, 2025, GitHub Repository. [Online]. Available: <https://github.com/ub4raf/Ninebot-PROTOCOL>
- [8] -, “Ninebot ES Communicaton Protocol”, 2019, Webpage. [Online]. Available: <https://cloud.scooterhacking.org/release/nbdoc.pdf>
- [9] -, “numpy.polyfit documentation”, 2025, Webpage. [Online]. Available: <https://numpy.org/doc/stable/reference/generated/numpy.polyfit.html>
- [10] C. Önen, A. Pandharipande, G. Joseph, and N. J. Myers, “Occupancy Grid Mapping for Automotive Driving Exploiting Clustered Sparsity,” *IEEE Sensors Journal*, vol. 24, no. 7, pp. 9240-9250, 2024. [Online]. Available: <https://doi.org/10.1109/JSEN.2023.3342463>.
- [11] D. Casado Herraez, M. Zeller, L. Chang, I. Vizzo, M. Heidingsfeld, and C. Stachniss, “Radar-Only Odometry and Mapping for Autonomous Vehicles,” *arXiv preprint*, 2023. [Online]. Available: <https://arxiv.org/abs/2305.12409>
- [12] S. R. Bhatt, B. S. Nadiger, R. Parthasarathy, and H. M. Shetty, “Instantaneous Ego-motion Estimation Using Doppler Radar,” *IEEE Sensors Letters*, vol. 7, no. 5, pp. 1–4, 2023. [Online]. Available: <https://doi.org/10.1109/LSENS.2023.3244030>
- [13] C. E. Beal, T. Williams, J. Pauli, M. Mukadam, and B. Boots, “Robust Off-Road Autonomy Using Multimodal Sensor Fusion,” in *Proc. of the Conference on Robot Learning (CoRL)*, 2023. [Online]. Available: <https://openreview.net/forum?id=kmiZqSgoAt>
- [14] B. Sundaralingam, C. E. Beal, and B. Boots, “Robust High-Speed State Estimation for Off-Road Autonomous Vehicles,” in *Proc. of Robotics: Science and Systems (RSS)*, 2023. [Online]. Available: <https://openreview.net/forum?id=3JpfLY3ihix>
- [15] GeeksforGeeks, *DBSCAN Clustering in ML — Density Based Clustering*, 2023. [Online]. Available: <https://www.geeksforgeeks.org/dbSCAN-clustering-in-ML-density-based-clustering/>. [Accessed: 19-Mar-2025].
- [16] MathWorks, *Understanding Kalman Filters, Part 3: Optimal State Estimator*, 2017. Available at: <https://la.mathworks.com/videos/understanding-kalman-filters-part-3-optimal-state-estimator--1490710645421.html> (Accessed: March 23, 2025).
- [17] Texas Instruments, *Radar Toolbox – mmWave Sensor Configuration and Demos*, 2024. Available at: https://dev.ti.com/tirex/explore/node?node=A__ADnbl7zK9bSRgZqeAxprvQ__radar_toolbox__1AslXXD__2.20.00.05 (Accessed: March 23, 2025).
- [18] Movella (Xsens), *MTi User Manual*, 2023. [Online]. Available: https://www.xsens.com/hubfs/Downloads/usermanual/MTi_usermanual.pdf. [Accessed: 26-Sep-2025].
- [19] Scottapotamas, *Xsens MTi Device Interface and Parser*, GitHub repository, 2020. [Online]. Available: <https://github.com/Scottapotamas/xsens-mti>. [Accessed: 26-Sep-2025].