Jornadas de Automática

Land–Maritime Localization in Autonomous Robotics: Nav2 versus EasyNav

Mejía, Irene[a,*], Angarita, Juan Camilo[a], Cervera, Enric[a], Marín, Raúl[b]

[a] *Robotic Intelligence Lab (RobInLab), Dept. of Computer Science and Engineering, Universitat Jaume I, Av. Vicent Sos Baynat, s/n, 12071 Castelló de la Plana, Spain.*
[b] *Interactive and Robotic Systems Laboratory (IRS-Lab), Dept. of Computer Science and Engineering, Universitat Jaume I, Av. Vicent Sos Baynat, s/n, 12071 Castelló de la Plana, Spain.*

To cite this article: Mejía, I., Angarita, J.C., Cervera, E., Marín, R. 2026. Land–Maritime Localization in Autonomous Robotics: Nav2 versus EasyNav. Jornadas de Automática, 47.
https://doi.org/10.17979/ja-cea.2026.47.13851

Resumen

Este trabajo analiza la localización LiDAR-inercial en plataformas robóticas terrestres y marítimas mediante una comparación entre Nav2 y EasyNav. Para ello, se emplea un módulo externo de percepción, compuesto por un LiDAR Livox MID-360 y una NVIDIA Jetson Orin Nano, común a ambas plataformas e independiente del sistema sensorial propio de cada robot. Los datos registrados se procesan con Fast-LIO2 para obtener una estimación odométrica y reconstruir nubes de puntos, que posteriormente se filtran, alinean y proyectan para generar representaciones útiles para la localización. La comparación se centra en cómo cada *framework* integra esta información perceptiva: Nav2 mediante representación *online* y `costmaps` 2D, y EasyNav mediante un mapa de ocupación previo. Los resultados en el escenario portuario muestran que EasyNav ofrece una localización más regular en posición, mientras que Nav2 resulta más sensible a la variabilidad de la percepción local.

Palabras clave: Robots móviles, Robótica de campo, Percepción y sensorización, Fusión de información y sensores, Guiado, navegación y control, Vehículos autónomos de superficie, Vehículos marinos no tripulados.

Land–Maritime Localization in Autonomous Robotics: Nav2 versus EasyNav

Abstract

This work analyzes LiDAR-inertial localization in terrestrial and maritime robotic platforms through a comparison between Nav2 and EasyNav. To this end, an external perception module, composed of a Livox MID-360 LiDAR and an NVIDIA Jetson Orin Nano, common to both platforms and independent of each robot's native sensing system, is used. The recorded data are processed with Fast-LIO2 to obtain an odometric estimation and reconstruct point clouds, which are subsequently filtered, aligned, and projected to generate representations suitable for localization. The comparison focuses on how each framework integrates this perceptual information: Nav2 through online representation and 2D `costmaps`, and EasyNav through a prior occupancy map. The results in the port scenario show that EasyNav provides more regular position localization, while Nav2 is more sensitive to the variability of local perception.

Keywords: Mobile robots, Field robotics, Perception and sensing, Information and sensor fusion, Guidance navigation and control, Autonomous surface vehicles, Unmanned marine vehicles.

1. Introduction

Robust localization is a fundamental component for the autonomous operation of mobile robots, as it directly affects tasks such as navigation, guidance, planning, and control. This problem is especially relevant in environments where GPS is not available, is not sufficiently accurate, or must be complemented with local perception, as occurs in indoor

*Autor para correspondencia: al415507@uji.es
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

spaces, port areas, or partially structured environments (Becoy et al., 2025).

LiDAR-inertial systems have gained a prominent role in this context, as they combine three-dimensional geometric information with high-frequency inertial measurements. Methods such as Fast-LIO2 make it possible to estimate sensor odometry and reconstruct dense point clouds of the environment, providing a useful perceptual basis for platforms with complex dynamics (Xu et al., 2022). However, the quality of this odometry alone does not guarantee robust localization, since the final outcome also depends on how each framework transforms, represents, and integrates this information (Gao and Awuah-Offei, 2026).

This issue is especially relevant when comparing platforms of different nature. In legged terrestrial robots, such as the Unitree Go2 Pro (Unitree Robotics, 2024), locomotion introduces oscillations and variations in the sensor pose (Brekke et al., 2026). In surface marine robots, water-induced motion and the availability of geometric references in port environments pose different challenges for maintaining a coherent pose with respect to the environment.

In ROS 2, Nav2 has become one of the most widely adopted reference frameworks for mobile navigation, with an architecture mainly based on two-dimensional maps and `costmaps` (Macenski et al., 2020, 2023). EasyNav, in turn, represents a more recent reference framework for modular navigation, providing a flexible architecture capable of working with 2D maps, `NavMaps` defined as surfaces in a three-dimensional space, and other 3D or hybrid representations (Martín Rico et al., 2025). This difference makes it possible to analyze how the same perceptual source can be integrated differently into the localization process.

This work compares LiDAR-inertial localization in terrestrial and marine robotic platforms using Nav2 and EasyNav. The objective is to investigate how the integration strategy adopted by each framework influences localization coherence when both rely on the same LiDAR-inertial perception source. The main contributions are: i) the integration of a reusable external LiDAR-inertial module; ii) the processing of its data to obtain representations suitable for navigation; and iii) the comparison of both frameworks in terms of localization coherence.

2. System Architecture

The experimental architecture is built around two robotic platforms: a terrestrial quadruped robot, the Unitree Go2 Pro (Unitree Robotics, 2024), and a catamaran-type autonomous surface vehicle. In both cases, an external perception module is used, consisting of a Livox MID-360 LiDAR-inertial sensor (Livox Technology, 2024), an NVIDIA Jetson Orin Nano processing unit (NVIDIA, 2023), and an independent power supply system (Table 1). This configuration makes it possible to decouple perception from the native sensing system of each robot and maintain a common LiDAR-IMU basis for the experiments with Nav2 and EasyNav.

2.1. Plataformas y módulo de percepción

The Unitree Go2 Pro represents the terrestrial case, where legged locomotion introduces body oscillations and variations in the sensor pose. The catamaran-type USV represents the marine case, affected by water-induced motion and by the availability of geometric references in the port environment. These differences make it possible to evaluate LiDAR-inertial localization on platforms with different dynamics (Figure 1).

((a)) ((b))

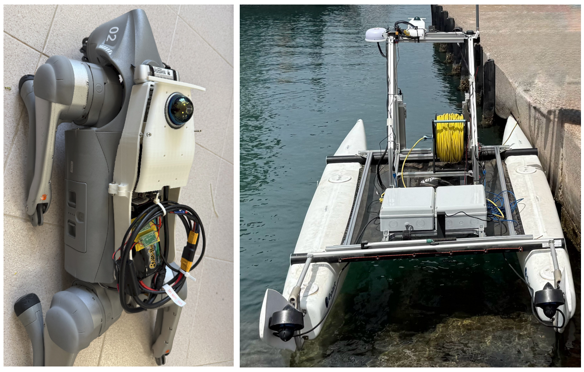


Figure 1: Integration of the external perception module on the two robotic platforms. (a) Unitree Go2 Pro. (b) Catamaran-type USV.

The structure used was previously developed in Angarita Noriega et al. (2025) as an external sensing support for mobile robotic platforms. In this work, it is reused as a common perception block, keeping the LiDAR, the processing unit, and the mounting configuration constant throughout the experiments. Although Nav2 and EasyNav rely on this same perceptual basis, the information is incorporated through different input pipelines and representations, which form the core of the comparison.

Employing a vehicle-independent provides a reusable and adaptable architecture. This approach facilitates the evaluation of different perception processing strategies and mechanical configurations across multiple robotic platforms while maintaining common experimental conditions. It also enables the perception pipeline to be validated first in controlled terrestrial environments before being transferred to more demanding maritime trials, while minimizing the influence of hardware-dependent sensing characteristics in the comparison between Nav2 and EasyNav.

Table 1: Summary of the experimental configuration used.

Category	**Component**	**Main use**
Platform 1	Unitree Go2 Pro	Quadruped robot
Platform 2	Robotic catamaran	USV surface robot
Computing	NVIDIA Jetson Orin Nano	External processing
Sensor	Livox MID-360	3D LiDAR + IMU
Middleware	ROS 2 Humble	System integration
Odometry	Fast-LIO2	LiDAR-inertial Odometry estimation
Processing	PCL[a]	Point cloud preprocessing
Map	Occupancy grid[b]	Global 2D representation
Navigation	Nav2 / EasyNav	Comparative analysis

[a]Functions used: *PointXYZI*, *SACSegmentation*, *PassThrough*, *transformPointCloud*.

[b]Obtained by projecting the filtered point cloud onto the XY plane through point accumulation per cell.

3. Environment Perception

This section describes the pipeline used to obtain a consistent representation of the environment and a localization estimate from the system sensor data. The process ranges from data acquisition and odometry estimation using Fast-LIO2 to filtering, alignment, and generation of a cartographic representation used as a reference in the experimental evaluation.

3.1. Data Acquisition

Data acquisition was carried out in two different experimental scenarios. The tests with the quadruped robot were conducted in a structured indoor environment, consisting of corridors, walls, and open areas, representative of buildings where GPS cannot be used. This scenario made it possible to analyze the continuity of the localization and the coherence of the generated representation in an environment with well-defined geometric references.

On the other hand, the tests with the USV were carried out in an outdoor port environment, with a larger scale and more variable perceptual conditions, including open spaces, maritime structures, and vessels located near the operating area. In addition, the presence of communication systems and equipment typical of port environments could introduce electromagnetic interference, reflections, or occasional signal degradation. This scenario made it possible to evaluate the system under conditions closer to a real maritime application.

The data were stored in different `rosbags2` in `.db3` format, mainly including the `/livox/lidar` (`CustomMsg`, 10 Hz) and `/livox/imu` (`Imu`, 200 Hz) topics, corresponding to the Livox LiDAR and its integrated IMU. These recordings were used as input to the localization and reconstruction pipeline based on Fast-LIO2.

3.2. Odometry Estimation and 3D Map Generation

The 3D map was generated from the LiDAR-inertial data processed using Fast-LIO2 (Xu et al., 2022). This method fuses LiDAR measurements with those from the inertial measurement unit (IMU) to incrementally estimate the motion of the sensor and register the point clouds in a common reference frame. As a result, both an odometric estimate of the platform and an accumulated point cloud of the environment were obtained.

The offline execution on the different `rosbags` made it possible to process the recorded sequences homogeneously. In the terrestrial case, the point cloud was exported directly in PCD format; whereas in the maritime scenario, due to the lower presence of stable geometric references, the partial point clouds were aligned using ICP before fusion, using parameters selected to ensure robust registration (Besl and McKay, 1992).

3.3. Point Cloud Filtering

The resulting point cloud was refined using the `Point Cloud Library` (Rusu and Cousins, 2011) before being projected onto the 2D plane. A ROS 2 node applies a `PassThrough` filter to the vertical coordinate, retaining

$$P_{\text{filtered}} = \{p \in P \mid p_z \leq z_{\max}\}$$

where P is the input point cloud and $z_{\max}$ is set to -0.5 m for the terrestrial scenario and -0.2 m for the maritime one. This preprocessing removes irrelevant structures and reduces spurious occupancies.

3.4. Offline Gravity-Based geometric alignment of the Point Cloud

The reconstructed point cloud may remain tilted with respect to the horizontal plane due to the relative orientation between the LiDAR, the external structure, and the reconstruction reference frame. To prevent this inclination from affecting the 2D projection, an offline ROS 2 alignment node was implemented on the PCD cloud. The node detects the dominant ground plane using RANSAC, restricting the search to approximately horizontal planes and computes a rotation to align it with the vertical axis.

This transformation is applied only to the point cloud used to generate the occupancy map, without modifying the odometry produced by Fast-LIO2.

3.5. Occupancy Map Generation

Once the 3D cloud had been filtered and aligned, the three-dimensional information was projected onto the horizontal plane to generate an occupancy map in `occupancy grid` format, used as the cartographic basis for navigation. The projection was performed with a resolution of 0.08 m/cell, as a trade-off between geometric detail, computational cost, and robustness against local noise.

Each cell was marked as occupied according to the number of projected points inside it. In the terrestrial map, obtained with the quadruped, a minimum threshold of one point per cell was used, whereas in the maritime scenario this value was increased to four due to the higher point density.

Although a variant based on 2D RANSAC was implemented to extract linear segments from the occupied cells, the results presented use the grid obtained directly through the accumulation of projected points.

4. Navigation Stacks

Before describing each configuration, Figure 2 summarizes the general integration workflow followed.

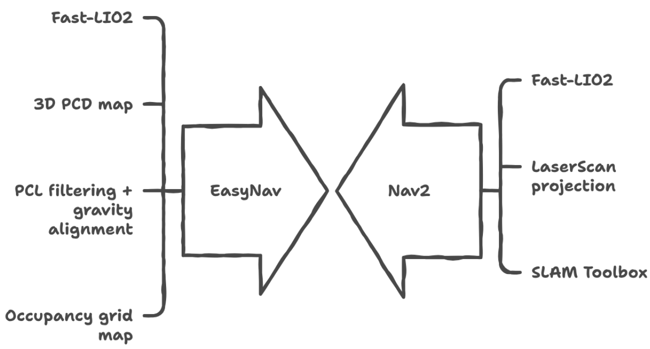


Figure 2: LiDAR-Inertial Odometry Integration Workflow.

During execution, the robot was teleoperated while the LiDAR-inertial estimate was recorded and its coherence with respect to the generated cartographic representations was analyzed. To this end, an additional transformation was incorporated between the initial frame of the estimator and an auxiliary frame aligned with gravity. This correction makes it possible to express the LiDAR observations in a reference system suitable for localization, without modifying the odometry produced by Fast-LIO2 or the PCD cloud generated offline.

4.1. Nav2

Nav2 was integrated into the robot through an unofficial SDK (Abizov, 2024), which adapts the communication interface of the Unitree Go2 Pro to the ROS 2 ecosystem and provides a functional configuration of the stack. In this work, Nav2 was used as a ROS 2 integration reference on the real platform during teleoperation.

The information from the LiDAR-inertial pipeline was incorporated by projecting the 3D point cloud onto the horizontal plane, generating a representation in LaserScan format. This input was integrated into the `costmaps` through `VoxelLayer` layers. The configuration used launched SLAM Toolbox in online mode together with the Nav2 navigation stack, including `SmacPlannerHybrid` as the global planner and `DWBLocalPlanner` as the local controller.

In this configuration, Nav2 does not directly use the occupancy map generated offline in Section 3, but instead follows its usual workflow. However, the generated map could be used in an alternative configuration through SLAM Toolbox in localization mode, avoiding the need to reconstruct the environment during the test. Therefore, the comparison with EasyNav also reflects the differences between an online generated representation and a previously built occupancy map.

4.2. EasyNav

EasyNav is a modular framework for ROS 2 aimed at decoupling the localization, planning, and control modules from the specific representation of the environment. This structure makes it possible to integrate different maps and perceptual sources without depending on a single cartographic model.

In this work, it was used as an alternative integration approach within the proposed experimental framework, using the occupancy map generated offline as the global reference representation. Unlike Nav2, it starts from a cartographic representation that has been previously filtered, corrected with respect to the vertical axis, and projected onto the horizontal plane.

During execution, this global representation is combined with local observations to maintain the robot localization with respect to the map. The configuration used included an AMCL localizer over a `costmap` and a `Serest` local controller, while keeping maximum velocities equivalent to those used in Nav2. In this way, EasyNav makes it possible to analyze an integration approach in which the cartographic information comes directly from the LiDAR-inertial reconstruction process, with less dependence on online mapping stages.

5. Experimental Evaluation and Results

The experimental evaluation analyzed the localization obtained by both stacks during the playback of data recorded with the teleoperated robot. Autonomous navigation was not evaluated; instead, the temporal evolution of the estimated pose and its coherence with the LiDAR-inertial perception were assessed.

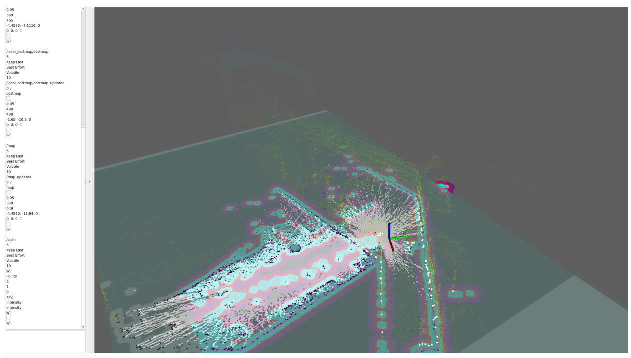
During execution, a transformation was added between the initial frame of the algorithm and an auxiliary frame aligned with gravity. This correction acts only on the relationship between reference frames, without modifying the Fast-LIO2 Odometry or the PCD cloud generated offline.

Therefore, the comparison should not be interpreted as an isolated evaluation of the localization algorithms under identical cartographic conditions, but rather as a comparative analysis of the integration workflows used by each stack within the considered experimental configuration.

5.1. Observed Results with Nav2

During the tests, Nav2 generated an online representation of the environment and maintained a continuous estimate of the robot pose throughout the analyzed segment. The recorded information made it possible to verify the update of the `costmaps` and the availability of a global pose associated with the map frame (Figure 3).

((a))



((b))

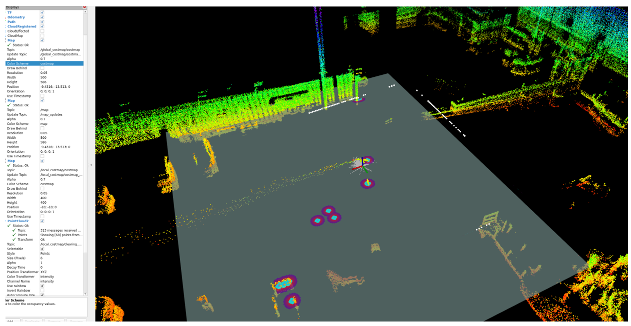


Figure 3: RViz visualization of the Nav2 representations. (a) Land environment. (b) Port environment.

However, as execution progressed, a gradual loss of coherence was observed between the LiDAR-inertial cloud generated by Fast-LIO2 and the representation built by Nav2. This degradation was more evident in the port scenario, where the lower presence of continuous geometric structure and the greater dispersion of the observations made it more difficult to maintain a stable correspondence between local perception and the map. Although the stack continued operating, the estimated pose showed greater spatial variability and slight discontinuities at certain moments.

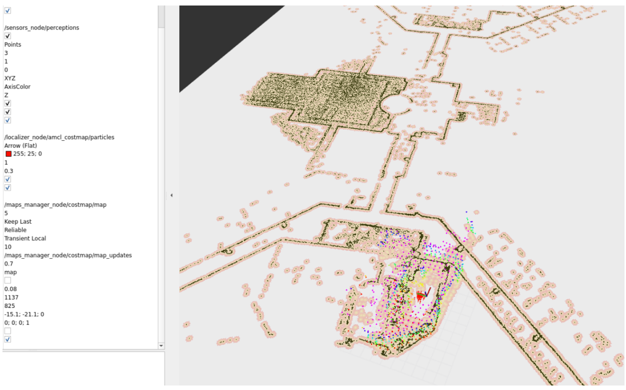
Overall, Nav2 showed functional behavior within a complete navigation stack configuration, although the localization exhibited greater spatial variability compared with that obtained using a prior cartographic reference.

5.2. *Observed Results with EasyNav*

In the case of EasyNav, the previously generated occupancy map was used as the global reference for localization, while local perception made it possible to incorporate information from the environment during execution. This combination is especially relevant in mobile platforms, where small variations in the pose or in the state estimation can affect the coherence between current observations and the cartographic representation.

Figure 4 shows the RViz visualization of the representations used by EasyNav in the terrestrial and port scenarios. In both cases, it can be observed how the global cartographic information is combined with local observations of the environment, making it possible to maintain a spatial reference during execution. In the terrestrial environment, the representation shows a more defined structure, associated with walls, corridors, and indoor obstacles. In the port scenario, the perception presents a more open and dispersed distribution, typical of an outdoor environment with less continuous geometric structure.

((a))



((b))

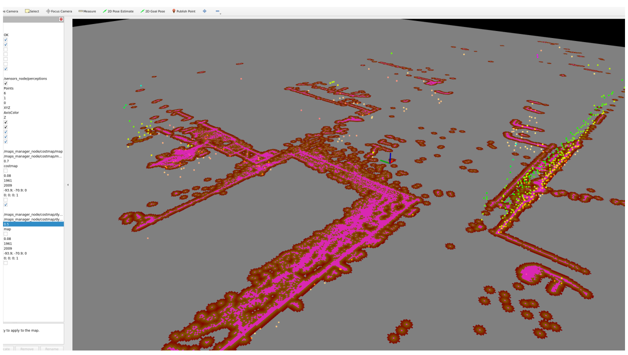


Figure 4: RViz visualization of the EasyNav representations. (a) Land environment. (b) Port environment.

During the experiments, EasyNav showed more regular localization when the estimated pose remained correctly aligned with respect to the global map. The use of a previously built cartographic representation reduced the dependence on online mapping, although the behavior remained conditioned by the correspondence between local observations and the reference map, especially in areas with noise, low point density, or partial loss of structure.

5.3. *Comparative Analysis of the Observed Behavior*

The results show clear differences in the way both stacks exploit the same LiDAR-inertial source. In Nav2, localization is more strongly conditioned by the representation generated during execution itself, so local irregularities in perception have a more direct effect on the estimated pose. This is especially noticeable in the port scenario, where the geometry is less structured and the observations appear more dispersed.

EasyNav, in contrast, starts from a previously built cartographic reference, which provides a more stable spatial basis for localization. This difference does not eliminate the dependence on local perception, but it reduces the influence of online mapping and makes it possible to maintain a more regular pose when the observations correctly match the global map.

To quantify these differences, the estimated poses were analyzed during the playback of the `rosbags`. The metrics in Table 2 correspond to the port scenario, as it was the case where the differences between both approaches were most noticeable. The LiDAR-inertial odometry from Fast-LIO2 was used as a relative reference, without being considered an absolute ground truth.

Table 2: Localization metrics obtained on the USV.

System	**XY RMSE [m]**	**Mean XY error [m]**	**Max. XY error [m]**
Nav2	0.0246	0.0141	0.1518
EasyNav	0.0090	0.0067	0.0328

Jumps > 0.10 **m**	**Max. jump [m]**	**Yaw RMSE [deg]**	**Max. Yaw error [deg]**
4	0.1405	0.421	2.998
0	0.0285	0.555	0.729

Both systems sustained a publication frequency near 20 Hz; thus, performance discrepancies are not attributable to pose update degradation. Notably, EasyNav exhibited superior relative positioning accuracy, yielding an XY RMSE of 0.0090 m versus 0.0246 m for Nav2. This metric underscores a more stable position estimate across the trajectory. In practical terms, this behavior reduces abrupt pose variations, facilitating localization in environments with sparse or irregular geometric features. Conversely, state estimation discontinuities —or "jumps"— indicate occasional changes in the estimated trajectory, which may affect localization stability during operation.

In terms of orientation, Nav2 obtained a slightly lower yaw RMSE, although EasyNav maintained a lower maximum angular error. This suggests that Nav2 achieved a slightly lower average yaw error, whereas EasyNav avoided larger angular deviations that could affect heading consistency. Overall, the results indicate that EasyNav provided more regular position localization in the evaluated port segment, while Nav2 maintained functional behavior but was more sensitive to local variations in the representation generated during execution.

5.4. *Supplementary Material*

The supplementary material associated with this work is available on Zenodo (Mejía Hurtado, 2026). It includes videos of the RViz visualization of the experiments conducted in the terrestrial and port scenarios, as well as representative fragments of the rosbags used for the evaluation. The generated maps, and the derived files used to obtain the localization metrics presented in this article are also included.

The aim of this material is to facilitate the visual inspection of the results and enable partial reproduction of the experimental workflow described, including the comparison between the online representation generated by Nav2 and the use of the offline occupancy map employed by EasyNav.

6. Conclusions and Future Work

The results obtained show that localization coherence does not depend solely on the quality of the available LiDAR-inertial perception, but also on the way each stack transforms and integrates this information. Although Nav2 and EasyNav used a common perceptual basis, their behavior differed due to the internal representations employed and the way each architecture manages the relationship between the map, local observations, and the estimated pose.

Nav2 enabled a functional integration of the LiDAR information, maintaining a workflow based on incremental mapping and costmaps. This approach facilitates compatibility with the standard ROS 2 ecosystem, but showed greater sensitivity to the quality of the 3D–2D projection and to the continuity of the observed geometry, especially in the port scenario.

EasyNav, in contrast, showed more regular position localization by using the previously generated occupancy map as a global reference. This cartographic representation reduced the dependence on online mapping and made it possible to exploit more directly the geometric structure obtained during offline processing. Nevertheless, its behavior remains conditioned by the correspondence between the prior map, local perception, and the estimated pose during execution.

The comparative analysis indicates that, in the evaluated port segment, EasyNav showed lower relative position deviation and fewer abrupt pose jumps, while Nav2 maintained functional behavior within its incremental mapping workflow without loading a prior map. These results should not be interpreted as an absolute measure of accuracy, since the LiDAR-inertial odometry from Fast-LIO2 was used as a relative reference, but rather as a comparison of localization stability and temporal coherence.

As future work, the evaluation will be extended to additional terrestrial and maritime segments, and alternative configurations of both stacks will be studied. In particular, it would be interesting to evaluate Nav2 using the previously processed occupancy map, analogously to EasyNav, through a map-based localization configuration. Additional tests with different map resolutions and occupancy thresholds could also be performed to analyze their influence on localization stability and on the coherence between local perception and the global representation. This would make it possible to determine whether the observed differences are mainly due to the use of online mapping, to the resolution and preprocessing of the cartographic representation, or to the internal architecture of each stack.

Acknowledgments

This work was made possible thanks to the facilities and resources provided by the RobInLab laboratory and the CIRTESU group at Universitat Jaume I, as well as the technical and academic support received during the experimental development. It was also developed within the framework of the research projects CENTAURO (PLEC2023-010360) and TANDEM (PID2023-149910OB-C32), funded by the Spanish Ministry of Science, Education and Universities.

References

Abizov, N., 2024. go2_ros2_sdk: Unofficial ros2 sdk for unitree go2. https://github.com/abizovnuralem/go2_ros2_sdk.

Angarita Noriega, J. C., Marqués Verdegal, J., Pino Jarque, A., Cervera Mateu, E., Marín Prades, R., 2025. Sistema embebido multimodal portable para navegación y mapeo. In: Jornadas de Automática. Vol. 46. DOI: 10.17979/ja-cea.2025.46.12213

Becoy, A. J., Khomenko, K., Peternel, L., Rajan, R. T., 2025. Autonomous navigation of quadrupeds using coverage path planning with morphological skeleton maps. Frontiers in Robotics and AI Volume 12 - 2025. DOI: 10.3389/frobt.2025.1601862

Besl, P. J., McKay, N. D., 1992. A method for registration of 3-d shapes. IEEE Computer Society 14 (2). DOI: 10.1109/34.121791

Brekke, V., Berge, E. O., Dybdahl, E., Singh, J., Tyapin, I., 2026. Ros 2-driven navigation and sensor platform for quadruped robots. Robotics 15 (4). DOI: 10.3390/robotics15040070

Gao, Y., Awuah-Offei, K., 2026. Efficient autonomous navigation of a quadruped robot in underground mines on edge hardware. URL: https://arxiv.org/abs/2603.04470

Livox Technology, 2024. Livox mid-360 lidar sensor. https://www.livoxtech.com/mid-360.

Macenski, S., Martin, F., White, R., Clavero, J. G., 2020. The Marathon 2: A Navigation System. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, p. 2718–2725. DOI: 10.1109/iros45743.2020.9341207

Macenski, S., Moore, T., Lu, D. V., Merzlyakov, A., Ferguson, M., 2023. From the desks of ros maintainers: A survey of modern map; capable mobile robotics algorithms in the robot operating system 2. Robotics and Autonomous Systems 168. DOI: 10.1016/j.robot.2023.104493

Martín Rico, F., Moreno Olivo, F. M., Manzanares Serrano, J. C., Guerrero Hernández, J. M., Cely Gutiérrez, J. S., Aguado, E., Romero Ramírez, F. J., 2025. Easynav simple stack: a minimum set of plugins for a reliable navigation. In: Herrero-Pérez, D., Martínez Barberá, H., Bernal Polo, P., Pavón Pulido, N. (Eds.), Proceedings of 25th International Workshop on Physical Agents (WAF25). Universidad Politécnica de Cartagena. DOI: 10.82432/10317/21021

Mejía Hurtado, I., 2026. Supplementary material for land–maritime localization in autonomous robotics: Nav2 versus easynav. DOI: 10.5281/zenodo.20564182

NVIDIA, 2023. NVIDIA Jetson Orin Nano developer kit. https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/nano-super-developer-kit/.

Rusu, R. B., Cousins, S., May 9-13 2011. 3d is here: Point cloud library (pcl). In: IEEE International Conference on Robotics and Automation (ICRA). Shanghai, China. DOI: 10.1109/ICRA.2011.5980567

Unitree Robotics, 2024. Go2 Pro robot. https://www.unitree.com/go2.

Xu, W., Lin, J., He, D., Zhang, F., 2022. Fast-lio2: Fast direct lidar-inertial odometry. IEEE Transactions on Robotics 38 (4), 2053–2073. DOI: 10.1109/TRO.2022.3141876