

Article

Forest Surveying with Robotics and AI: SLAM-Based Mapping, Terrain-Aware Navigation, and Tree Parameter Estimation[†]

Lorenzo Scalera^{1,*}, Eleonora Maset¹, Diego Tiozzo Fasiolo¹, Khalid Bourr², Simone Cottiga²,
Andrea De Lorenzo², Giovanni Carabin³, Giorgio Alberti⁴, Alessandro Gasparetto¹, Fabrizio Mazzetto³
and Stefano Seriani²

¹ Polytechnic Department of Engineering and Architecture, University of Udine, 33100 Udine, Italy; eleonora.maset@uniud.it (E.M.); diego.tiozzo@uniud.it (D.T.F.); alessandro.gasparetto@uniud.it (A.G.)

² Department of Engineering and Architecture, University of Trieste, 34127 Trieste, Italy; khalid.bourr@dia.units.it (K.B.); simone.cottiga@phd.units.it (S.C.); andrea.delorenzo@units.it (A.D.L.); sseriani@units.it (S.S.)

³ Faculty of Agricultural, Environmental and Food Sciences, Free University of Bozen-Bolzano, 39100 Bolzano, Italy; giovanni.carabin@unibz.it (G.C.); fabrizio.mazzetto@unibz.it (F.M.)

⁴ Department of Agricultural, Environmental, and Animal Sciences, University of Udine, 33100 Udine, Italy; giorgio.alberti@uniud.it

* Correspondence: lorenzo.scalera@uniud.it

[†] This paper is an extended version of our paper published in Scalera, L.; Tiozzo Fasiolo, D.; Maset, E.; Carabin, G.; Seriani, S.; De Lorenzo, A.; Alberti, G.; Gasparetto, A. Mobile Robotics for Forest Monitoring and Mapping Within the AI4FOREST Project. In: Parikyan, T., Sargsyan, Y., Ceccarelli, M. (eds) *Mechanical Engineering Solutions: Design, Simulation, Testing, Manufacturing*. MES 2025. Mechanisms and Machine Science, vol 191, Springer, Cham, 2026.

Abstract

Forest surveying and inspection face significant challenges due to unstructured environments, variable terrain conditions, and the high costs of manual data collection. Although mobile robotics and artificial intelligence offer promising solutions, reliable autonomous navigation in forest, terrain-aware path planning, and tree parameter estimation remain open challenges. In this paper, we present the results of the AI4FOREST project, which addresses these issues through three main contributions. First, we develop an autonomous mobile robot, integrating SLAM-based navigation, 3D point cloud reconstruction, and a vision-based deep learning architecture to enable tree detection and diameter estimation. This system demonstrates the feasibility of generating a digital twin of forest while operating autonomously. Second, to overcome the limitations of classical navigation approaches in heterogeneous natural terrains, we introduce a machine learning-based surrogate model of wheel–soil interaction, trained on a large synthetic dataset derived from classical terramechanics. Compared to purely geometric planners, the proposed model enables realistic dynamics simulation and improves navigation robustness by accounting for terrain–vehicle interactions. Finally, we investigate the impact of point cloud density on the accuracy of forest parameter estimation, identifying the minimum sampling requirements needed to extract tree diameters and heights. This analysis provides support to balance sensor performance, robot speed, and operational costs. Overall, the AI4FOREST project advances the state of the art in autonomous forest monitoring by jointly addressing SLAM-based mapping, terrain-aware navigation, and tree parameter estimation.

Keywords: artificial intelligence; environmental perception; forest inspection; forest surveying; mobile robotics; navigation challenges; terrain traversability



Academic Editor: Jiangjian Xiao

Received: 22 December 2025

Revised: 13 January 2026

Accepted: 14 January 2026

Published: 14 January 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Home to roughly 80% of all living species, forests cover about 30% of the Earth's land surface. The United Nations Environment Programme reported that between 1990 and 2020 the planet lost 4.4% of its forested area, around 178,000 km², approximately equivalent to the size of Libya [1]. This issue has been specifically addressed in the Sustainable Development Goals (SDGs) of the 2030 Agenda for Sustainable Development of the United Nations. For instance, SDG 15 (Life on Land) calls for the protection and sustainable management of forests, the expansion of forested areas, and appropriate funding for their stewardship. Monitoring forest environments is essential for safeguarding ecosystems and for fighting climate change but is also crucial for addressing other environmental challenges, including fire prevention, water flow regulation, and flood and drought mitigation [2].

A forest inventory involves the systematic collection of data on trees, e.g., species, diameter, height, age, vitality, and site conditions, in order to characterize the forest and monitor its evolution over time. Traditionally, forest surveys rely mainly on visual assessments and basic instruments. However, Unmanned Aerial Vehicle (UAV) surveys and satellite imagery are currently widely employed for this purpose [3]. In addition, terrestrial surveys conducted beneath the canopy offer highly detailed and accurate information. These are typically performed by trained operators equipped with modern sensing technologies, including 2D [4] and 3D laser scanners [5], Global Navigation Satellite System (GNSS) receivers, and advanced algorithms for automated tree species recognition [6]. In recent years, research has increasingly focused on the development of the forest digital twin (DT): a virtual, three-dimensional replica of the forest that integrates comprehensive information about each individual tree [7].

Over the past several years, mobile robotics has increasingly been adopted for surveying and environmental monitoring, with applications ranging from the mapping of hazardous or cluttered areas to the inspection of buildings [8], archaeological sites [9], and agricultural fields [10]. Compared with human-led surveys, robots can provide measurements that are more objective, consistent, and repeatable. Moreover, ground robots provide a complementary perspective to that of UAVs, enabling the integration of close-range, high-resolution data with aerial observations [11]. Robotic navigation typically involves three main components: perception, localization, and path planning [12]. GNSS-based techniques offer reliable global positioning in open outdoor spaces. However, their performance drops significantly where satellite visibility is obstructed or signal quality is degraded, e.g., under dense forest canopies or in heavily wooded terrain. In such conditions, autonomous platforms rely primarily on terrestrial sensing and Simultaneous Localization and Mapping (SLAM) algorithms, implemented using Light Detection and Ranging (LiDAR) or visual sensors [2]. These algorithms allow the robot to build a representation of its surroundings while estimating its position within that map. Nonetheless, achieving robust autonomy in forests remains a major challenge [13].

In the forestry context, mobile robotics is progressively being adopted to address a variety of complex and labor-intensive tasks, such as ecosystem monitoring, wildfire management, forest inventory, planting, pruning, and harvesting [14–16]. Early approaches investigated the use of legged robots in wooded areas, as reported in [17,18]. The majority of robotic systems employed in forestry environments rely on teleoperation and do not exhibit full autonomy in navigation. A tracked mobile robot capable of computing an optimal sequence of way points for navigation in forest-like settings is presented in [19]; nevertheless, this method assumes a structured arrangement of trees in rows, which rarely reflects the irregular nature of real forests. Reliable identification of traversable terrain remains a fundamental requirement for autonomous forest robots, as it enables safe motion,

efficient path planning, and obstacle avoidance [20]. In this context, the work presented in [21] introduces a navigation and exploration framework for wheeled mobile robots operating in unknown forest environments, leveraging Gaussian process models to efficiently estimate free space. A self-supervised learning strategy for predicting traversable paths in forest scenarios is proposed in [13]. Similarly, the approach in [22] exploits visual features extracted from RGB images to improve global path planning by estimating traversability costs. Despite demonstrating autonomous navigation capabilities, the majority of these robotic systems are not specifically designed to generate detailed digital forest models that contain information such as tree positions, diameter at breast height (DBH) or other dendrometric parameters.

Several approaches have been proposed to improve tree detection and the automatic extraction of tree parameters from forest data. Recent advances in artificial intelligence (AI) have enabled accurate segmentation of trees from dense point clouds, effectively handling occlusions and the high variability in tree morphology [23]. In this context, PointNet++ [24] has been widely adopted for point-wise semantic labeling and segmenting tree trunks and branches [25] or to classify forest point clouds into stems, vegetation and terrain [26]. Likewise, U-Net-based encoder–decoder architectures extended to 3D through sparse convolutions have been successfully applied to forest tree segmentation tasks [27,28]. On the other hand, deep learning approaches operating on visual data acquired by mobile robots have been employed to identify tree trunks at the image level. For instance, a multimodal trunk detection approach combining visible and thermal imagery is presented in [29], whereas [30] uses YOLOv7 [31] to estimate tree locations and build a 2D map of the environment. Additionally, Cascade Mask R-CNN [32], pre-trained on synthetic data and fine-tuned on real forest images, is used in [33] for trunk segmentation. However, several systems, such as [34], are limited to surface classification and therefore do not support complete forest inventory generation. To the best of the authors' knowledge, the development of a fully autonomous ground robot able to navigate complex forest environments, the path planning of the robot based on terrain state and wheel–soil interaction, and the optimal estimation of tree parameters still represent open research problems.

To solve the challenges mentioned above, the PRIN 2022 project titled “An Artificial Intelligence Approach for Forestry Robotics in Environment Survey and Inspection (AI4FOREST)” aims to design and develop an autonomous robotic system capable of navigating through forest environments to create a detailed digital twin of the natural landscape. The project described in this paper focuses on the following key elements:

- The development of an autonomous robotic system designed to monitor forest areas, and its application to gather detailed environmental data.
- The development of a terrain-aware navigation controller tailored for unstructured forest environments, combining an AI-based decision model with a SLAM algorithm for robust autonomous operation.
- The investigation of the minimum point cloud density required to accurately extract tree parameters (e.g., diameter and height).

This paper is an extended version of our previous conference publication in [35]. With respect to [35], in this work: (i) we extend the description of the autonomous navigation framework for the mobile robot in forest; (ii) we introduce and validate a machine learning (ML)-based surrogate model framework for wheel–soil interaction, enabling dynamics simulation and traversability-aware path planning for rough terrains; and (iii) we present a study on determining the minimum and optimal cloud point densities necessary for accurately estimating tree diameter at breast and tree height.

The paper is organized as follows: Section 2 describes the materials and methods used in the project. More in detail, Section 2.1 illustrates the development of the autonomous mobile robot for forest surveying, Section 2.2 presents the ML-based simulation and traversability-aware navigation framework, and Section 2.3 presents the analysis on the impact of point cloud density on tree parameter estimation. The results achieved in the AI4FOREST project are described in Section 3, and discussed in Section 4. Finally, the conclusions of this work and future research directions are outlined in Section 5.

2. Materials and Methods

2.1. Development of an Autonomous Mobile Robot for Forest Surveying

This section focuses on the first challenge addressed by the project: enabling autonomous navigation of a mobile robot in forest environments and generating a detailed 3D reconstruction of the surveyed area, including the automatic detection of trees and the estimation of their structural parameters. The proposed robotic system is built on an AgileX Scout 2.0 platform and integrates a LiDAR sensor, an RGB camera, an IMU, wheel-encoder odometry, and a GNSS receiver that supplements localization whenever satellite signals are available (Figure 1). The combined weight of the onboard devices remains under the robot maximum payload capacity, equal to 50 kg, and does not noticeably impact the performance of the wheel motors compared to when the robot is operating without any payload. The main characteristics of the onboard sensing modules are reported in Table 1. The LiDAR device used onboard the Scout 2.0 mobile robot is a 3D sensor and its vertical field of view is $\pm 15^\circ$ (30°). This allows the robotic system to reconstruct the trees in their complete height.

The key contribution of the proposed framework lies in the integration of measurements from multiple onboard sensors within a custom SLAM framework. This framework enables the robot to accurately localize itself and autonomously navigate densely vegetated terrain while simultaneously constructing a three-dimensional model of the environment. In addition, a tailored integration of perception modules is introduced to analyze the reconstructed scene. Trees are automatically identified through a deep-learning vision pipeline and a complementary LiDAR-based spatial clustering, which jointly support the estimation of trunk diameter and other relevant attributes. A schematic overview of the complete architecture is presented in Figure 2, and the main functional blocks are discussed in detail in Sections 2.1.1 and 2.1.2.

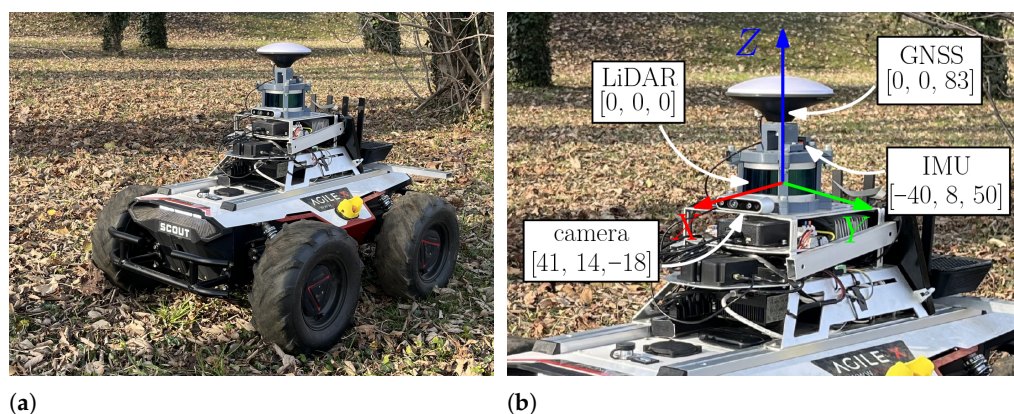
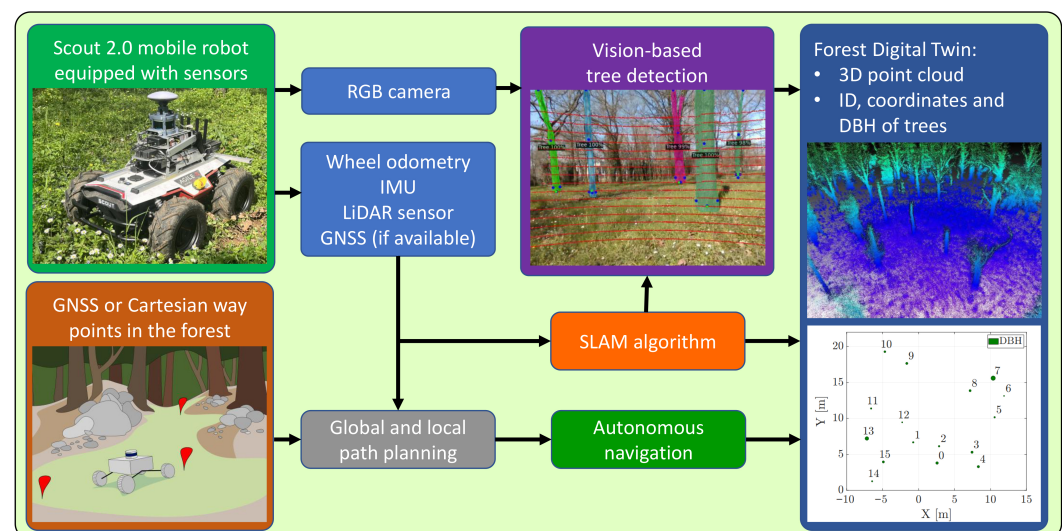


Figure 1. The autonomous mobile robot (a); onboard sensors and relative positions of the sensors with respect to the robot frame in millimeters (b). The LiDAR sensor reference frame is assumed to be coincident with the robot reference frame.

Table 1. Platform and onboard sensors used for forest surveying.

Device	Model	Technical Specifications
Mobile robot	Scout 2.0 (AgileX, Shenzhen, China)	Weight: 62 kg; length: 930 mm; width: 699 mm; height: 348 mm.
Computer	Jetson AGX Xavier (NVIDIA, Santa Clara, CA, USA)	GPU: 512-core NVIDIA Volta architecture; CPU: 8-core NVIDIA Carmel Arm v8.2 64-bit CPU 8MB L2 + 4MB L3; OS: Ubuntu 18.04; ROS Melodic.
RGB camera	RealSense D435 (Intel, Santa Clara, CA, USA)	Frame resolution: 1920 × 1080 pixel; frame rate: 30 fps.
LiDAR sensor	VLP-16 (Velodyne, San Jose, CA, USA)	Channels: 16; measurement range: 100 m; range accuracy: up to ±3 cm; FoV (vertical): ±15° (30°); FoV (horizontal): 360°; rotation rate: 10 Hz.
IMU	MTi-630 (Xsens, Enschede, The Netherlands)	Sensor fusion accuracy: 0.2° roll/pitch, 1° heading; gyroscope noise density: 0.007°/s/√Hz; accelerometer noise density: 60 µg/√Hz.
GNSS receiver	SimpleRTK2B Budget kit (Ardusimple, Andorra la Vella, Andorra)	U-blox ZED-F9P module; precision: ≤1 cm with NTRIP; update rate: max. 10 Hz; first RTK fix: 35 s.

**Figure 2.** Overview of the proposed architecture for forest monitoring and mapping.

2.1.1. Autonomous Navigation in Forest

An autonomous navigation system specifically optimized for forest applications is developed to allow the mobile robot to survey and inspect challenging wooden environments. As it moves through the environment, the robotic platform collects the data required for 3D reconstruction, tree detection, and estimation of the diameter at breast height (DBH). A summary of the algorithms adopted in the experimental campaign is presented in Table 2, together with their corresponding input and output. The navigation strategy relies on either GNSS waypoints defined in the WGS84 reference system or, in scenarios where satellite signals are degraded by dense canopy cover, on Cartesian goals expressed in the local East-North-Up (ENU) frame. Precise localization of the robot is achieved by combining information from multiple onboard sensors, namely LiDAR, IMU, wheel encoders, and GNSS. Sensor data is integrated through the LIO-SAM SLAM framework, which

enables simultaneous real-time mapping and pose tracking while ensuring high-quality 3D reconstruction, even under challenging conditions such as uneven terrain or wheel slippage [36]. Leveraging multi-sensor fusion, the system effectively compensates for the weaknesses of individual measurements, including errors introduced by variable ground traction, thus avoiding the need for an explicit model of terrain–tire friction [37]. Both localization and map generation are performed directly in full 3D, allowing the system to naturally operate in complex terrains characterized by slopes and surface irregularities.

The initialization of a local fixed ENU reference frame derived from IMU measurements is required before autonomous navigation can start. As the robot traverses the environment, the LIO-SAM algorithm progressively refines the 3D map by integrating successive LiDAR scans. In the time intervals between acquisitions, the robot pose is predicted through an Extended Kalman Filter that fuses IMU readings with wheel encoder data. To account for the accumulation of drift over time, the system further exploits the absolute positioning information from GNSS, when available, together with a loop-closure strategy based on the Euclidean distance between points.

The LIO-SAM SLAM algorithm was selected for its capability to effectively fuse information from multiple odometry sources into a coherent and robust state estimate, thus compensating for the limitations of single sensors. By combining LiDAR, inertial, and wheel-based measurements, the system achieves improved accuracy and reliability in complex forestry scenarios. In addition, when compared to alternative mapping strategies, LIO-SAM is able to produce a point cloud of the environment that is more dense, consistent, and better suited for 3D reconstruction and metric analysis.

The proposed navigation strategy operates without the need for a pre-existing map of the environment. At the outset, the trajectory between each pair of consecutive way points is simply defined as a straight-line segment. This choice relies on the assumption that trees represent localized obstacles that can be effectively handled by the collision avoidance system embedded in the robotic platform. As a result, global path planning is kept deliberately simple, while obstacle detection and avoidance are managed locally in real time during navigation. More in detail, the robot generates reference trajectories as straight-line segments connecting successive way points using the Carrot Planner algorithm [38]. This global path planner receives a target position defined by the user and verifies whether the specified goal lies within an obstacle. If this is the case, the algorithm iteratively moves the goal backward along the direction vector connecting the robot to the original target until a collision-free point is identified. This adjusted target is then forwarded to the robot local planner for execution. Through this mechanism, the Carrot Planner defines the global path as a sequence of straight-line segments connecting consecutive way points, enabling the robot to approach each user-defined goal as closely as possible while maintaining safety.

Obstacle avoidance is handled through dynamically updated cost maps built in real time from the most recent LiDAR measurements. For local trajectory refinement, the system adopts the Timed Elastic Band (TEB) algorithm, which continuously reshapes the planned path during execution [39]. By optimizing the trajectory with respect to traversal time, safety margins from obstacles, and kinematic constraints on velocity and acceleration, TEB enables the robot to move efficiently while ensuring safe operation. The two-dimensional cost maps generated online encode the traversability of the surrounding areas and support real-time safe navigation. Although forest environments often feature uneven terrain, the use of a 2D representation for obstacle avoidance proves to be a reasonable approximation in many practical situations, as the robot mainly operates on fairly gradual slopes.

Table 2. Algorithms used for the experimental tests and corresponding input and output.

Function	Algorithm	Input	Output
SLAM	LIO-SAM [36]	Wheel odometry, IMU, LiDAR, GNSS data	Point cloud, robot pose
Global path planning	Carrot Planner [38]	GNSS or Cartesian way points	Global path
Local path planning	TEB [39]	Global path	Robot velocity commands
Tree detection	PercepTreeV1 [33]	Camera images	Tree mask, 2D keypoints
Clustering	DBSCAN [40]	3D keypoints	DBH, tree coordinates

2.1.2. Tree Detection and Diameter Estimation

Tree detection, localization, and DBH estimation are achieved through a multi-sensor perception pipeline that integrates camera imagery, 3D LiDAR scans, and SLAM-based robot poses. The overarching objective is to obtain reliable, spatially consistent measurements of individual trees while operating in unstructured forest environments.

The pipeline begins with tree segmentation and keypoint extraction performed in RGB images by the PercepTreeV1 deep learning architecture, built upon Mask R-CNN [33]. The model outputs bounding boxes, segmentation masks, and structural keypoints, i.e., trunk edges and trunk center. PercepTreeV1 is adopted due to its demonstrated robustness and generalization ability across heterogeneous forest conditions beyond those seen in its training set, which is essential for autonomous field deployment.

To transition from 2D to 3D tree positions and to recover metrically accurate tree traits, LiDAR measurements are subsequently incorporated. Leveraging the extrinsic calibration between the LiDAR and RGB camera, 3D LiDAR points are projected onto the image plane, following an approach similar to [41]. Figure 3 illustrates an example of tree detection, with projected LiDAR points shown in red and keypoints representing trunk edges and centers in blue. The keypoints predicted in the image are associated with LiDAR points through a nearest-neighbor search, thereby assigning 3D coordinates to each detected trunk feature. A distance-based filtering step discards detections beyond 10 m, where both LiDAR density and network prediction reliability tend to degrade, ensuring that only high-confidence measurements contribute to the estimation process. The reconstructed 3D trunk edges are then used to infer DBH, while the trunk center yields the 3D tree position in the LiDAR coordinate frame. These estimates are transformed into the global reference frame by exploiting the robot pose provided by the SLAM algorithm, thereby ensuring spatial consistency across the entire mapped area.

Because individual trees may be observed multiple times from different viewpoints, a subsequent data-association stage consolidates repeated detections. A density-based clustering method (DBSCAN [40]) groups geometric estimates belonging to the same physical tree, rejects outliers, and assigns a unique tree identifier. For each resulting cluster, the final trunk position and DBH are computed as the median of all associated measurements, providing robustness against occasional perceptual noise. The entire estimation pipeline is executed offline, allowing the integration of multiple observations and ensuring high-fidelity reconstruction of tree features.

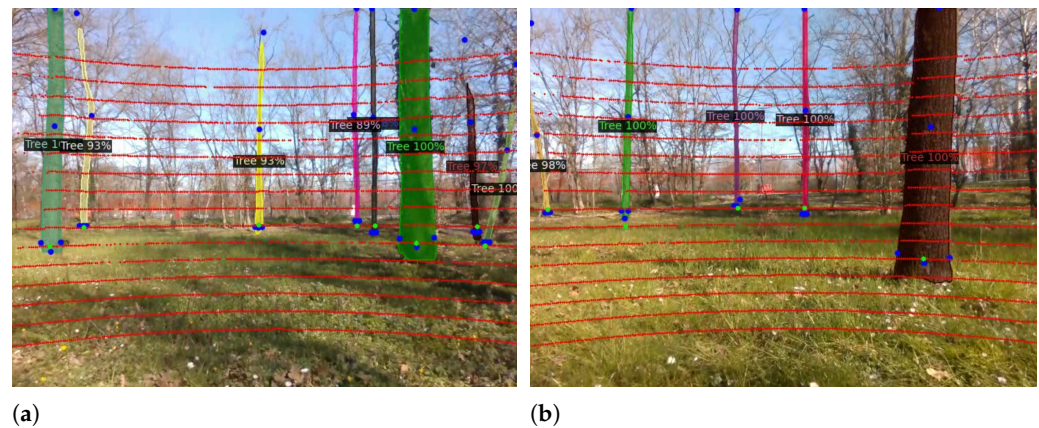


Figure 3. Examples of the outputs of the tree-detection pipeline in the image plane (a,b). The segmentation masks and keypoints (blue) predicted by the PercepTreeV1 network are shown together with the LiDAR points projected into the image (red). The various trees are detected with high confidence by the deep learning architecture.

2.2. Simulation Framework for Terrain-Aware Navigation

A key challenge in autonomous forest navigation is predicting robot traversability on soft, heterogeneous soils where wheel–soil interactions, such as slip and sinkage, may significantly affect mobility. Traditional planners ignore such effects, leading to infeasible paths and hence unreachable destinations. To address this, we propose a physics-based simulation framework grounded in Bekker-Wong terramechanics theory, combined with a machine learning surrogate model that enables traversability prediction. In this context, the term *terrain-aware* refers specifically to the local, physics-informed prediction of wheel–soil interaction properties such as slip and sinkage, rather than to a holistic, *environment-aware* global navigation strategy. While the proposed framework can be integrated into a global planner, the core contribution lies in modeling and predicting local traversability based on soil mechanics. This allows us to capture realistic wheel–soil behavior while keeping computation fast enough for online path planning.

The ability to simulate the terrain where the robots operate is important for two main reasons: (a) it enables training ML-based models, (b) it allows us to do performance testing on the resulting control algorithms. At the core of our simulation architecture, stands a ML-based model that emulates the interaction between a robot wheels and the soil during driving. As shown in Figure 4, the model itself has a dual use: it can be the base for a local or global path planner, and be the surrogate model for wheel–soil interaction for dynamics modeling. In principle, the model accepts the control signals (commanded wheels rotation speed $v_{c,i}$ and steering angles α_i), the terrain characteristics (soil type, slope, slope direction), and outputs the effective wheel translation velocities $v_{p,i}$.

The surrogate model (SM) used in this work is based on the XGBoost regression [42], a gradient-boosted decision tree framework, chosen for its ability to capture complex nonlinear relationships and computational efficiency. We implement a split architecture with separate models for longitudinal (v_{xg}) and lateral (v_{yg}) velocity predictions, since each component is governed by a distinct physical mechanism. The predicted feed into the control module (CM) to enable simulation of the dynamics of wheel–soil interaction, as explained in [37], or to the path planner after processing the predictions to provide traversability information.

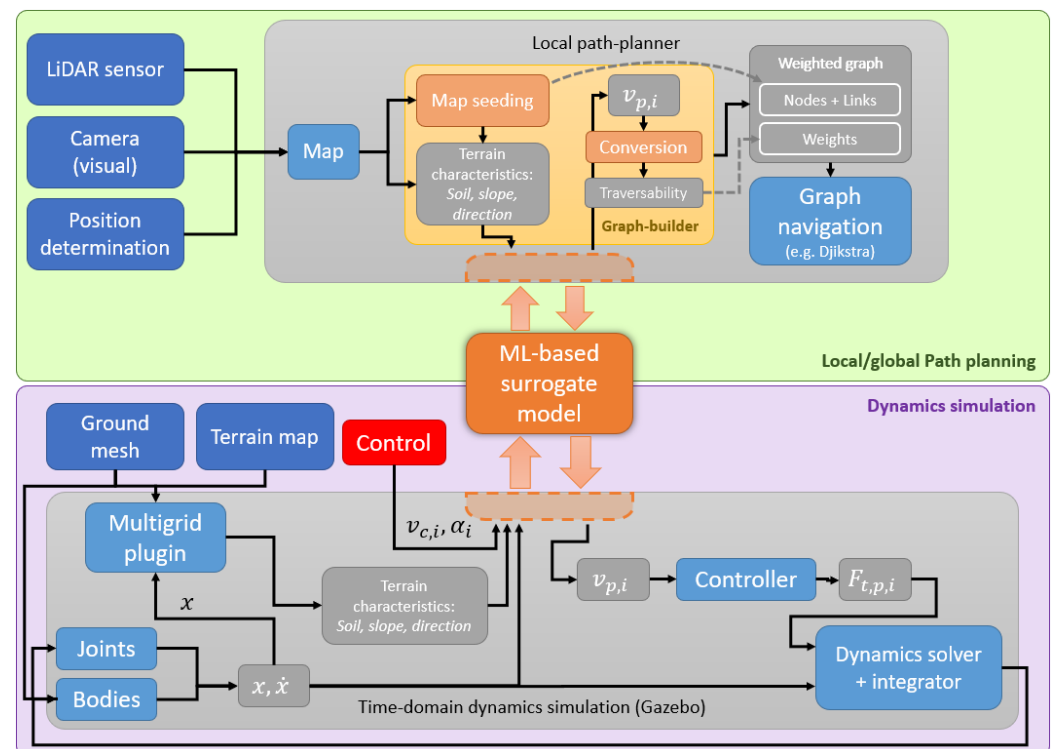


Figure 4. The dual-use diagram of the ML-based surrogate model: for local/global navigation, and for wheel–soil interaction dynamics simulation.

Gazebo Classic [43] is chosen as dynamic simulator. In order to obtain a realistic robot behavior on rough terrains, such as that of a forest, though, its native wheel–soil contact model needs to be enhanced. A custom Gazebo plugin has been used for the purpose, which implements a terramechanic model based on advancements of the classic Bekker’s theory [44,45].

The robot used for the simulations is the model of the Archimede rover, a 5 kg robotic system designed and produced at the Department of Engineering and Architecture of the University of Trieste, and now part of the Space Robotics Laboratory of the Department. The rover was validated both dynamically [46] and in terms of its implementation in Gazebo [47]; similarly, it was used to validate an earlier iteration of the ML-model [37]. Figure 5 shows the Archimede rover used for the simulations.

Based on the data collected from this plugin, a synthetic dataset for the SM training is generated, as shown in Figure 6. A tilted floor has been used as the base environment, with a slope sweeping from 2.5° to 15° with a 2.5° step, and four possible terrain types. The robot has been driven in straight lines, with a commanded velocity of 0.1 m/s to 1 m/s with a 0.1 m/s step, and an approach angle that sweeps from -180° to 180° with a 22.5° step. Overall, more than 3800 simulations were run and, for each of them, data from all the 4 robot wheels were collected. The final synthetic dataset is composed by 91 million data points, where the input features are terrain type, terrain slope, commanded wheel velocity, wheel approach angle and wheel load, while the target outputs are the components of the effective wheel velocity.

Building on the generated dataset, we trained both models using 20 estimators with maximum depth of value 6 to balance the complexity of the model with computational efficiency for our large-scale dataset and prevent overfitting. We adopted an 80–20 train-test split and relied on histogram-based trees with GPU acceleration to speed up training. Both models employ physics-informed features to capture velocity-angle effects, the influence of wheel loads, and terrain–slope interactions to improve prediction accuracy.

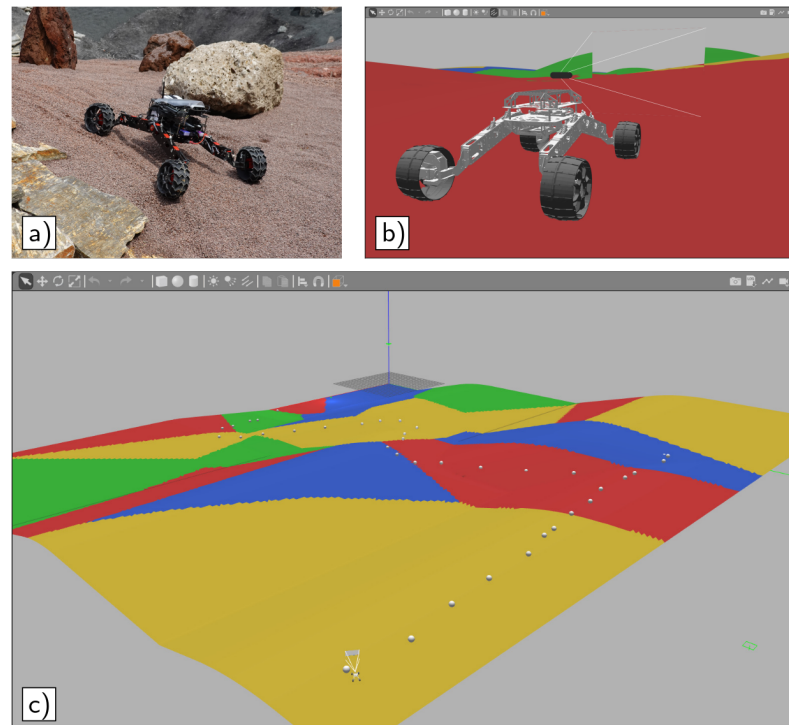


Figure 5. The Archimede rover platform used for the simulations: (a) the rover in a rough terrain environment; (b) Gazebo model; (c) typical simulation run.

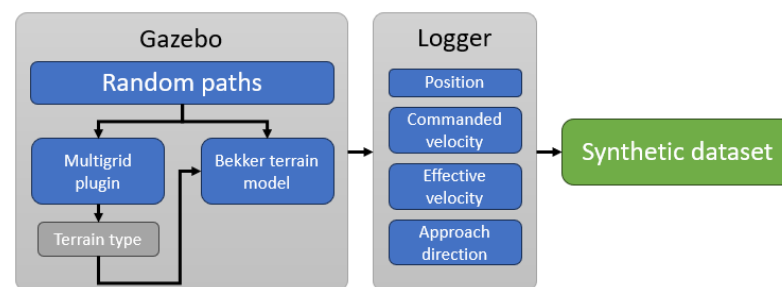


Figure 6. Generation of the synthetic dataset for the ML-based surrogate model.

2.2.1. Traversability-Based Path-Planning

In order to find the most traversable path between two points in a known environment, one should consider both the characteristics of the terrain, and the capabilities of the robot. In particular, the soil type, the slope, and the approach direction are important aspects that drive a measure of how traversable is a patch of terrain. As for the robot, its specific kinematics can hinder its aptitude at following some paths, e.g., sharp turns. As such, we have developed a path-planning methodology that leverages a graph-based navigation where edges are weighted based on their traversability.

More specifically, this approach can be used in both the case where the map is known in advance, and when the map is either partially or fully unknown. In the former, as shown in Figure 7, the surrogate model is used for local navigation, i.e., within line-of-sight from the robot, and possibly for already explored areas; in the latter, the model can be used for the whole map. In both instances, the underlying mechanism is that of using graph-based navigation (e.g., using a Dijkstra algorithm), where the weights of the graph edges are computed to reflect the traversability of the terrain between the edge nodes. Each edge is sampled at regular intervals between nodes, such that there are $k_{sub-sample}$ equally spaced sub-samples.

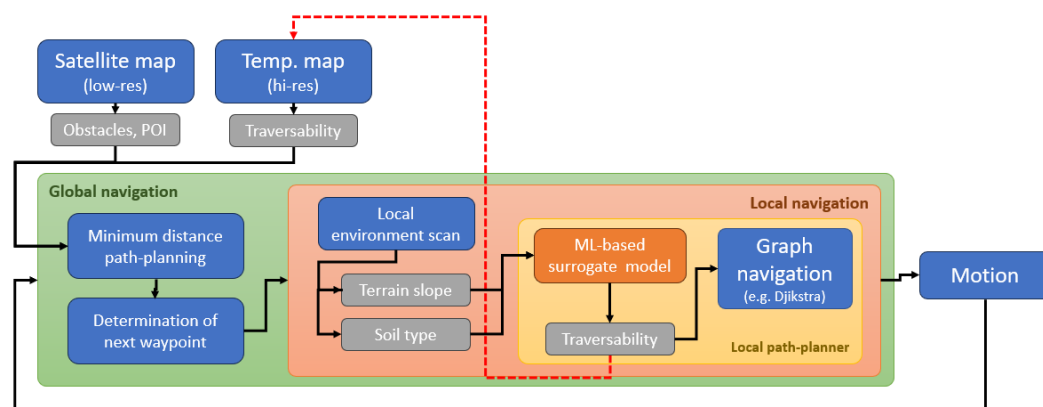


Figure 7. Diagram for the global and local navigation logic.

Traversability is based on the magnitude of sliding while driving; this is modeled as follows, using Euclidean norms $\|\dots\|_2$,

$$T(x_{ss}, y_{ss}) = 1 - \frac{\|(\mathbf{v}_{cmd} - \mathbf{v}_{pred})\|_2}{\|\mathbf{v}_{cmd}\|_2} \quad (1)$$

where x_{ss}, y_{ss} are the sub-sampling points along the relevant edge where the velocities are predicted by the ML-model; next, $\mathbf{v}_{cmd} = v \cdot (\cos \theta, \sin \theta)$ represents the commanded velocity vector, and $\mathbf{v}_{pred} = (v_{xg}, v_{yg})$ is the machine learning-predicted actual velocity. In principle, T ranges from 0 (low traversability) to 1 (high traversability), excluding edge-cases, e.g., negative velocities. The weight computation then combines three key terms: a traversability cost W_{trav} , a path length $L_{i,j}$, and a turn penalty P_{turn} for each edge connecting nodes i and j . The total edge cost is computed as follows:

$$C_{i,j} = k_1 W_{trav}(i, j) + k_2 L_{i,j} + k_3 P_{turn}(i - 1, i, j) \quad (2)$$

where k_1, k_2 , and k_3 are weighting coefficients (typically set to 1.0, 1.0, and 0.2) that control the relative importance of terrain traversability, path length, and directional smoothness, respectively. The traversability cost W_{trav} is computed as follows, depending on the sampling method:

$$W_{trav}(i, j) = \begin{cases} 1 - \sum_{ss=1}^{k_{sub-sample}} T(x_{ss}, y_{ss}) / k_{sub-sample}, & \text{average method} \\ 1 - \min(T(x_{ss}, y_{ss})), & \text{minimum method} \end{cases} \quad (3)$$

$L_{i,j} = \|p_j - p_i\|_2$ is the Euclidean distance between nodes positions, and $P_{turn}(i - 1, i, j)$ is a cost that penalizes sharp directional changes, defined as $P_{turn}(i - 1, i, j) = |\theta_{out} - \theta_{in}|$, where $\theta_{in} = \arctan 2(y_i - y_{i-1}, x_i - x_{i-1})$ and $\theta_{out} = \arctan 2(y_j - y_i, x_j - x_i)$ represent the incoming and outgoing path angles, respectively. Non-traversable edges are marked by an arbitrarily large weight of 10^6 to effectively exclude them from optimal paths.

2.2.2. Soil Interaction Dynamics Simulation

The model was integrated into Gazebo through the architecture described in [37]. During simulation, a ROS node performs ML inference and publishes the predicted wheel velocities. A custom Gazebo plugin receives these predictions as target velocities and apply forces via a PID controller to match the simulated wheel motion to the predicted behavior. This replaces the native Gazebo interaction forces tangential to the ground plane, while the reaction force normal to the interaction plane is kept.

2.3. Impact of Point Cloud Density on Tree Parameter Estimation

Although numerous tools and libraries are available for extracting forest attributes from LiDAR point clouds, the accuracy of these analyses largely depends on the quality and density of the acquired data. A key challenge in laser scanning is that precise estimation of forest structural parameters requires highly dense point clouds. Achieving such density typically involves either advanced and costly acquisition systems with higher sampling rates and resolutions or reducing the rover speed, both of which increase the survey time and demand more accurate positioning. To address this trade-off, this section investigates the impact of point cloud density, derived from Mobile Laser Scanning (MLS) acquisitions, on the accuracy of extracting key forest inventory metrics such as DBH and tree height (TH). While previous studies have explored optimal density requirements for Terrestrial Laser Scanning (TLS) using both single- and multi-scan approaches [48,49], this aspect has not yet been systematically investigated for MLS systems. Starting from several high-density datasets available online, the point cloud density was progressively and artificially downsampled. At each step, core tree features were extracted using the 3DFin tool library [50], and the impact on result accuracy was evaluated.

2.3.1. Forest Datasets

Two different open-source datasets acquired using a Mobile Laser Scanning (MLS) system (i.e., GeoSLAM Zeb-Horizon) and available online were considered in this study. To reduce computational demands, each dataset was divided into smaller sub-units for individual processing.

- Forest 1 (subsets A, B, and C): this dataset [51] was collected in southern Finland, within the Evo region (61.19° N, 25.11° E). The site consists of a naturally managed boreal forest characterized by mixed-species stands dominated by Scots pine (*Pinus sylvestris* L.), Norway spruce (*Picea abies* (L.) H. Karst.), and broadleaved species such as silver birch (*Betula pendula* Roth), downy birch (*Betula pubescens* Ehrh.), and European aspen (*Populus tremula* L.).
- Forest 2 (subsets C, D, and E): this dataset [50] is openly accessible via the 3DFin platform. It includes approximately 43.76 million points covering a total area of 684 m². Unfortunately, no reference information is available regarding the forest's location or characteristics.

Both models refer to flat forests typical of Finland. While this work focuses on generic forest environments, these datasets provide a solid starting point for defining the acquisition system minimum specifications (i.e., minimum point cloud density). Table 3 reports the main characteristics of the various subplots.

Table 3. Summary of forest plots and their associated attributes.

Plot Characteristics	Unit	Forest 1			Forest 2		
		Plot A	Plot B	Plot C	Plot D	Plot E	Plot F
Surface area	[m ²]	228	228	228	228	228	228
Point cloud	[M points]	15.97	8.94	7.55	3.35	15.97	16.40
Density	[points/m ³]	2234	1282	1064	588	1810	3677
Number of trees		18	14	17	9	13	11
DBH mean	[m]	0.19	0.21	0.18	0.23	0.19	0.20
DBH max	[m]	0.25	0.27	0.24	0.49	0.25	0.22
DBH min	[m]	0.13	0.12	0.10	0.17	0.11	0.17
TH mean	[m]	13	12.40	12.90	22.03	18.31	18.44
TH max	[m]	15.15	14.06	14.77	24.53	20.38	20.42
TH min	[m]	11.23	10.22	10.58	20.46	14.88	17.05

2.3.2. Tree Features Recognition Process

The analysis of the point cloud and the identification of different forest features were carried out using the 3DFin library [50] and CloudCompare software [52]. The point cloud was processed following a structured workflow consisting of three main steps, based on the approach presented in [53].

1. **Point Cloud Normalization.** Terrain effects were removed by subtracting the ground profile (Digital Terrain Model, DTM) from off-ground points, ensuring that the base of all trees was aligned to the same reference height. The DTM, with a spatial resolution of 0.45 m, was generated using the Cloth Simulation Filter (CSF) algorithm, a built-in function in CloudCompare. This algorithm is particularly effective in forested environments, as it filters out non-ground elements such as shrubs and stones, even under complex topographic conditions like uneven or sloped terrain [54].
2. **Individual Tree Identification.** A horizontal strip was extracted from the point cloud within the height range of 0.3 m to 5 m. All points within this interval were voxelized and grouped based on vertical continuity to identify individual stems and, consequently, trees.
3. **Tree Feature Measurement.** For the DBH estimation, points around 1.3 m height were extracted, and a circle was fitted using a nonlinear least-squares method that minimizes geometric error. If artifacts or irregularities were detected, the algorithm applied segmentation and iterative fitting to refine the diameter. TH was determined as the elevation of the highest point within the tree's influence area. This area was defined by clustering and voxelizing the point cloud around each identified stem, after removing outliers, using the DBSCAN algorithm.

Further details on the methodology and the algorithms used are provided in [55].

2.3.3. Reduction of MLS Point Cloud Density

In this work, point cloud density is defined as the number of points per cubic meter (points/m³) across the spatial extent of the dataset region. Before calculating this value, all non-forest elements, such as ground vegetation, stones, roads, buildings, and moving objects, were filtered out through a manual segmentation in CloudCompare. The spatial extent of each dataset was determined using the dimensions of an axis-aligned bounding box ($\Delta X, \Delta Y, \Delta Z$), calculated from the minimum and maximum coordinates. Table 3 presents the calculated density values for each subplot across the two forest sites. Starting from the original full-density point cloud, the various subplots were progressively thinned (i.e., point density reduction) using the Random Sampling by Percentage tool available in CloudCompare software. In particular, subsets were generated by sequentially retaining 90% to 10% of the original points in 10% increments, followed by additional subsets ranging from 9% to 1% in 1% increments (i.e., 19 distinct point cloud versions for each subplot). Importantly, each reduced subset was derived directly from the original full-density point cloud rather than from previously thinned versions. This approach ensured that each subset maintained a consistent and unbiased random distribution of points.

2.3.4. Performance Metrics

To comprehensively evaluate the performance of forest tree attribute detection, standard metrics and benchmarks were employed, including completeness, Root Mean Square Error (RMSE), and bias. Completeness estimates the proportion of correctly detected trees relative to a reference dataset and is calculated as:

$$\text{completeness} = \frac{\text{number of detected trees}}{\text{number of actual ground truth trees}} \times 100 \quad (4)$$

The RMSE provides an indication of the magnitude of error in the estimates, while bias quantifies systematic error between observed and actual values: bias can be positive (overestimation), negative (underestimation), or zero (no systematic error). In the absence of field-verified ground truth data (i.e., manual survey of each tree TH and DBH), the high-density (100%) point cloud was used as the reference dataset for comparative evaluation against the lower-density scenarios. Starting from this high-density point cloud, all the reference tree features reported in Table 3 were computed.

3. Results

This section presents the results of the project, structured around its three core activities: (i) first, the development and deployment of a sensorized robotic system equipped with LiDAR, IMU, GNSS, and camera, designed to monitor and acquire detailed data from a real forest environment; (ii) second, the implementation of an autonomous navigation controller tailored for unstructured forest terrain, which integrates an AI-based decision model with a SLAM algorithm to enable reliable robot localization and path planning; (iii) third, the investigation of the minimum point cloud density required to accurately extract tree parameters (e.g., diameter and height). This last contribution is preliminary for the post processing of the data acquired by the mobile robot, and the subsequent development of a virtual forest model to simulate the robot behavior during the development and testing of the navigation controller the creation of a virtual forest model that facilitates simulation-based testing and refinement of the navigation controller throughout the development process. Together, these components demonstrate the capabilities of integrating mobile robotics and artificial intelligence in forest surveying and inspection.

3.1. Experimental Results Obtained with the Autonomous Mobile Robot

The proposed forest monitoring and mapping approach using the Scout 2.0 mobile robot was evaluated in a wooded area of Cormor Park in Udine, Italy (Lat: 46°05′04.9″, Lon: 13°11′23.9″) under clear, sunny conditions. Before starting the experimental test, four way points were provided to the robot to define a 10 × 10 m square path. During the experiment, the robot traveled a total distance of 42.98 m in 107 s, achieving an average speed of 0.4 m/s. The effectiveness of the experimental results with the autonomous mobile robot is demonstrated through measurable indicators, including: successful completion of the autonomous mission, real-time obstacle avoidance enabled by global path replanning, SLAM map consistency, and accurate estimation of tree locations and DBH.

Figure 8a shows the 3D point cloud collected during the experiment, with points color-coded according to their elevation. The image in Figure 8a was captured in JRC 3D Reconstructor, which allows the user to visualize entire point clouds from any custom view. Individual trees can be readily distinguished thanks to the characteristic spatial distribution and density of the returns. The clear delineation of trunks and major branches attests to the quality of the data acquisition process and the effectiveness of the SLAM algorithm, confirming its suitability for tasks such as forest mapping and structural analysis. Figure 8b further reports the set of detected trees after DBSCAN clustering, each annotated with its unique identifier and estimated DBH. The experimental results indicate that the PercepTreeV1 model consistently identified tree stems with high reliability, particularly for trees located within 10 m of the robot. The subsequent DBSCAN-based association step successfully consolidated multiple observations of the same stem, yielding a single, coherent estimate of each tree position.

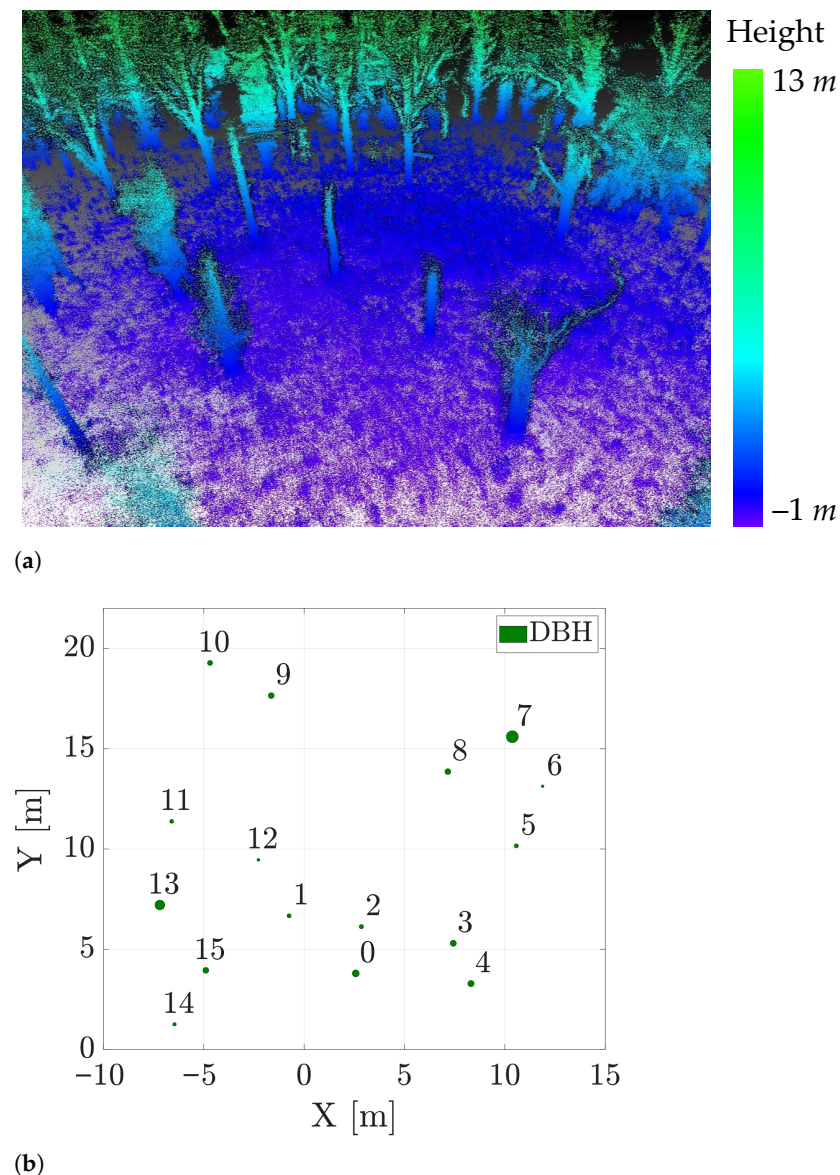


Figure 8. A 3D view of the point cloud acquired during the experiment, colored according to the point height (a); detected trees with assigned IDs, where the circle diameter is proportional to the estimated DBH (b).

3.2. Results of the ML-Based Framework for Terrain-Aware Navigation

In the following, we present the results of the machine learning-based traversability model we used both to simulate the rover driving and to perform the path-planning. First, we describe the *naked* performance of the model, comparing its predictions to the baseline; then, we present the results of the path-planning for two case studies intentionally designed to stress the model. Key performance metrics of the ML-based traversability model include predicted slip, path cost weighted by expected terrain interaction, and the ability to select safer and more traversable routes in heterogeneous terrains.

The methodology outlined in Section 2.2 led to the generation of a very large dataset, consisting of more than 70 million datapoints in the training-set and 18 million in the test-set. This was used to effectively train the XGBoost ML model, achieving remarkably good results. Figure 9 shows the correlation plots between the baseline and the predicted values for the v_{xg} and v_{yg} velocities. With a coefficient of determination of $R^2 = 0.9974$, the distribution shows very low dispersion, with very minor presence of outliers.

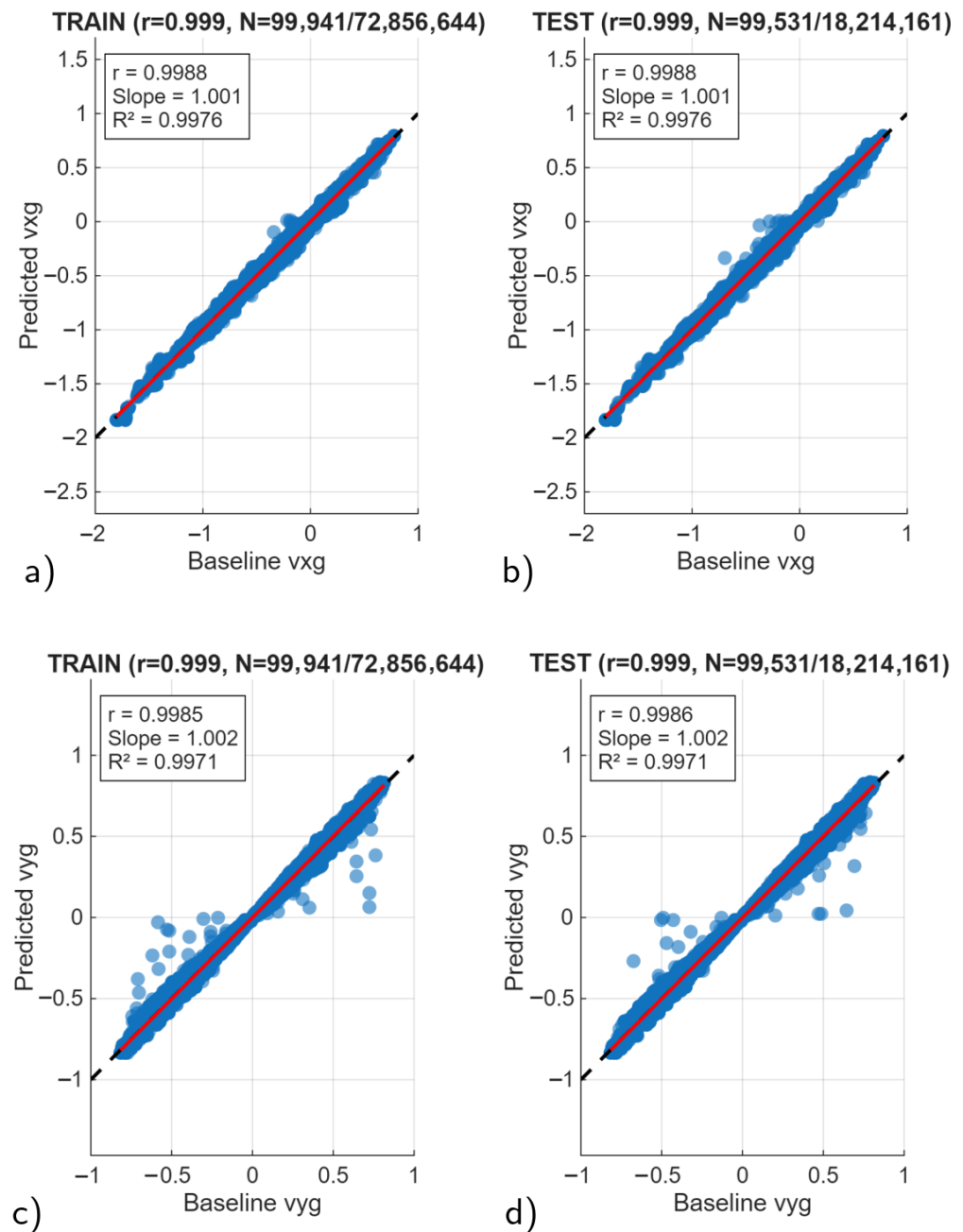


Figure 9. Performance of the ML-model for terrain-aware navigation: scatter plots of the baseline vs. predicted values of the v_{xg} (a,b) and v_{yg} wheel velocities (c,d). The dataset is decimated by a factor of approximately 1/150, to allow figure generation.

The traversability-based path planning architecture was applied to a set of two case studies: a tiered landscape, and a hilly environment (Figure 10). The *minimum* method described in Equation (3) was used to compute a path between two points of the maps. In particular, for both maps the starting point is [50,5] m, while the target stands at coordinates [95,95] m. Both maps are challenging for the models, being designed to severely limit straightforward paths, and offering a wide range of different traversability levels that need to be overcome in order to produce an optimal path. Both include non-traversable areas, both in terms of soil type and slope magnitude.

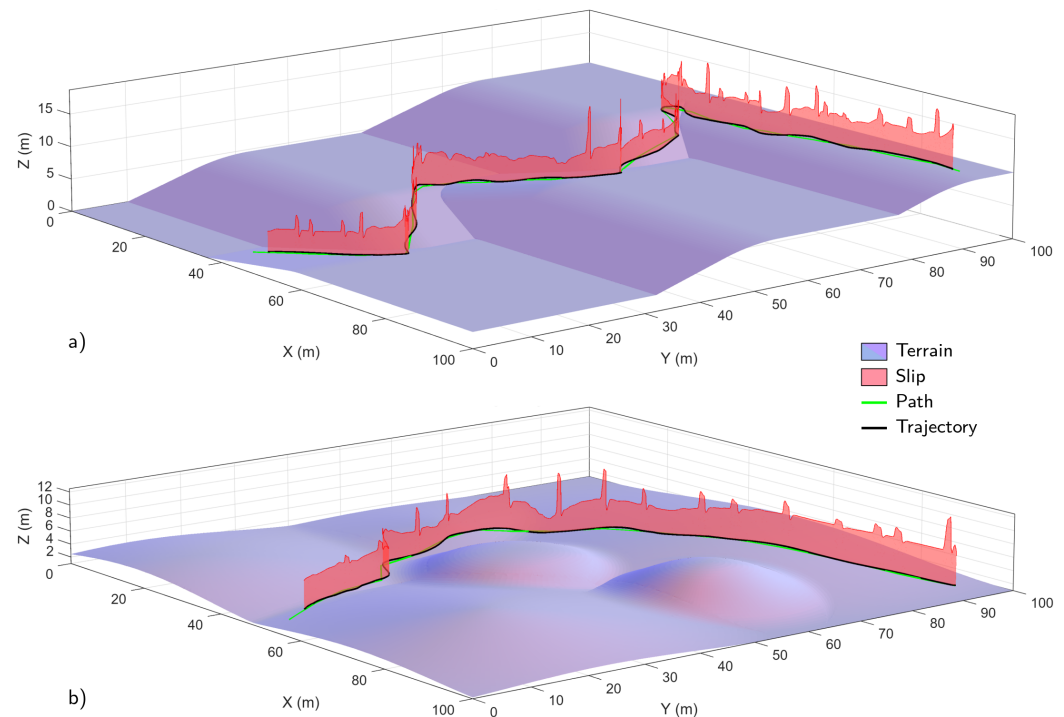


Figure 10. Results of the simulation in the Gazebo environment of two full runs using the traversability-based path-planner: (a) tiered landscape scenario; (b) hills scenario. In both, the red ribbon shows the profile of the slip during the driving. The green line shows the commanded path, whereas the black line indicates the actual driven trajectory.

3.3. Results of the Impact of Point Cloud Density on Tree Parameter Estimation

The results on point cloud density impact in tree parameter estimation identify explicit density thresholds for reliable DBH and tree height estimation, using RMSE and bias as evaluation metrics. The first result concerns the ability to detect trees as a function of point cloud density. Figure 11a shows that, on average across all subplots, more than 90% of trees are correctly detected when density exceeds 500 points/m³. At densities above 1000 points/m³, all trees are successfully identified.

Figure 11b,c present benchmarks for DBH (Diameter at Breast Height) estimation in terms of RMSE and bias. At low point cloud densities (<200 points/m³), RMSE exceeds 20 cm due to incomplete stem profiles and noise affecting circle fitting. As density increases, errors decrease rapidly: RMSE falls below 2 cm (approximately 10% of the average DBH) at around 600–700 points/m³, and stabilizes below 1 cm for densities ≥1000 points/m³. Bias patterns closely follow RMSE trends, with positive values indicating slight DBH overestimation. For densities above 600 points/m³, bias remains below 0.5 cm. This overestimation likely results from residual bark roughness and partial occlusions that shift circle fits outward.

Figure 11d,e show benchmarks for tree height (TH) estimation. RMSE decreases from 2.73 m at low densities to 0.18 m at the highest tested density. Errors drop sharply up to approximately 300 points/m³, then stabilize below 1 m (about 5% of the average tree height, 15–20 m). Bias remains low for densities above 300 points/m³, typically within ±0.2 m. The slight positive bias observed in some cases suggests a tendency to overestimate tree heights, possibly due to misclassification of outlier points (e.g., small branches or noise above the true apex) as part of the treetop.

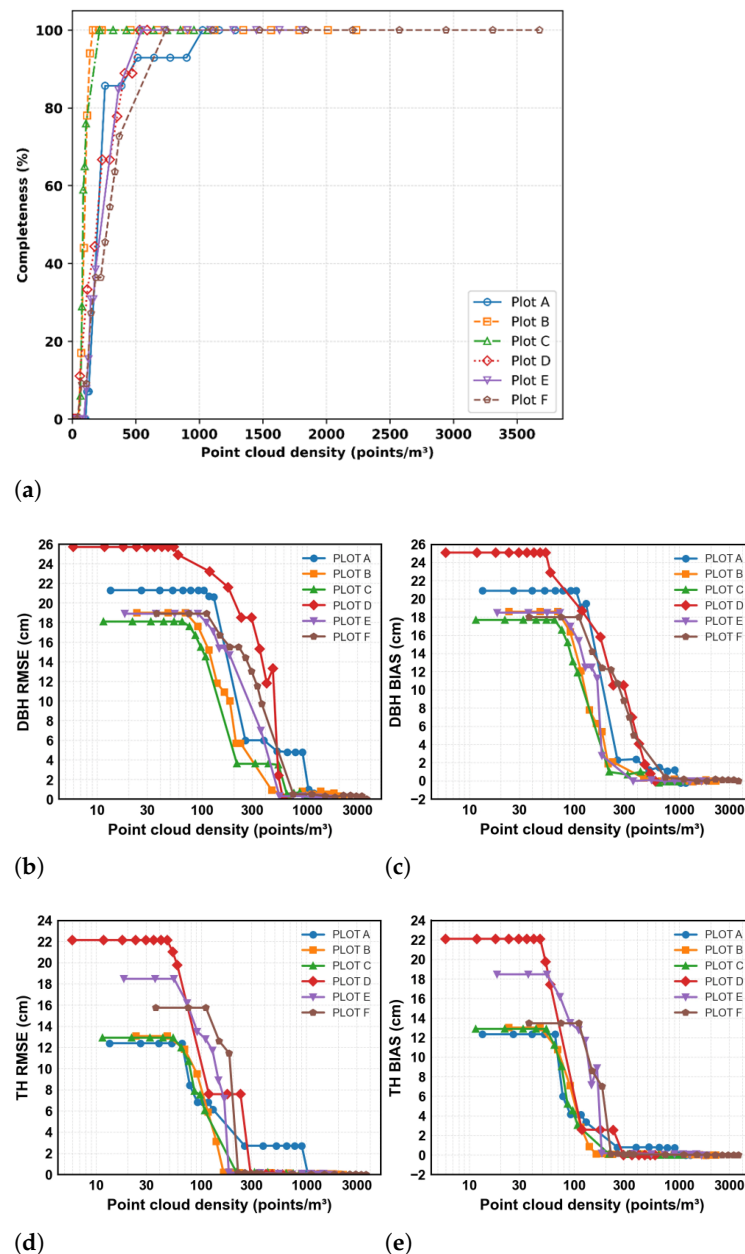


Figure 11. Results of the impact of point cloud density on tree parameter estimation: (a) completeness, (b) DBH RMSE, (c) DBH bias, (d) TH RMSE and (e) TH bias with respect to the cloud point density.

4. Discussion

The experimental results obtained with the SLAM-based autonomous mobile robot directly address one of the core challenges in forest surveying and inspection: achieving reliable autonomous navigation and data acquisition in cluttered, unstructured environments. By integrating SLAM-based localization, 3D mapping, and vision-based tree detection into a single robotic framework, the proposed system demonstrates that a ground robot can autonomously traverse forest environments while simultaneously generating a consistent map and extracting relevant forest parameters, such as tree locations and DBH. The experimental validation confirms that dynamic global path replanning enables effective obstacle avoidance in real time, while the combined LiDAR–camera perception framework supports accurate tree detection and spatial estimation. These results represent a concrete step forward compared to conventional manual or semi-autonomous forest surveys, which typically rely on predefined paths and lack continuous perception-driven adaptation.

At the same time, the experiments also highlight limitations that define clear directions for further improvement. Mapping accuracy and downstream estimation performance are strongly influenced by the robustness of the SLAM loop closure mechanism, sensor noise in the point cloud data, and the precision of LiDAR–camera calibration. In addition, the reliability of DBH estimation depends on the accurate extraction of trunk keypoints and on clustering strategies used to aggregate diameter and position measurements over time. These findings emphasize that, while the proposed system effectively solves the problem of autonomous data acquisition in forests, further advances in perception robustness and map consistency are required to improve the precision and repeatability of forest parameter estimation.

Regarding robot terrain-aware navigation, the proposed simulation framework tackles a well-known limitation of state-of-the-art forest robots: the lack of terrain-aware planning that accounts for soil–wheel interaction effects. The ML-based surrogate model, trained on a large synthetic dataset generated from classical Bekker terramechanics, provides a physically grounded solution that bridges high-fidelity dynamics simulation and traversability-aware path planning. Experimental validation in simulation demonstrates that the same model can be consistently used both to emulate realistic wheel–soil dynamics in Gazebo and to predict slip for navigation purposes. By weighting graph edges according to expected slip, the planner effectively selects paths that maximize traversability while accounting for terrain heterogeneity, slope, and motion direction. This unified modeling approach represents a meaningful advancement over purely geometric or kinematic planners, which neglect terrain mechanics and often fail in soft or uneven forest soils.

Nevertheless, an important limitation of the proposed mobility framework lies in the exclusive use of synthetic data for training the surrogate model. While the experimental results confirm strong adherence to the underlying physical model and significant computational advantages, real-world soil variability, such as moisture content, vegetation cover, and compaction, may not be fully captured. This highlights the need for future experimental campaigns to incorporate real field data to validate and refine the model, thereby reducing the sim-to-real gap and improving robustness during real deployments.

Finally, regarding the estimation of tree parameters, the analysis of point cloud density explicitly addresses the trade-off between sensing resolution, survey efficiency, and estimation accuracy, which is a critical yet often underexplored issue in robotic forest inventories. The results identify quantitative density thresholds for reliable DBH and tree height estimation, demonstrating that DBH accuracy is more sensitive to point density than height estimation. The observed differences between forest types further confirm that structural complexity and canopy gaps significantly affect estimation robustness. Although the study is limited by the absence of absolute ground-truth field measurements, the relative evaluation against high-density MLS data provides experimental evidence consistent with prior state-of-the-art studies and offers practical guidelines for sensor configuration and mission planning. At the same time, the exclusion of additional forest attributes, such as biomass or canopy metrics, defines a clear limitation and motivates future work toward more comprehensive forest characterization.

5. Conclusions

This paper presented the results of the AI4FOREST project, demonstrating the feasibility and effectiveness of integrating autonomous mobile robotics and artificial intelligence for forest surveying and inspection. The proposed robotic system combines SLAM-based navigation, 3D reconstruction, and deep learning techniques to generate digital twins of forest environments while accurately detecting trees and estimating their diameters.

Experimental results confirm the robot ability to autonomously navigate complex wooded terrains and to perform reliable mapping and data acquisition. Furthermore, the project introduced an ML-based surrogate model for wheel–soil interaction, enabling realistic simulation and traversability-aware path planning in heterogeneous natural terrains. Trained on synthetic data derived from classical terramechanics, the terrain-aware navigation model improves navigation robustness by exploiting predicted slip to select paths that maximize traversability across rough geometries. However, the sim-to-real gap remains a challenge since the surrogate model was trained entirely on synthetic data, and real-world soil variability (e.g., moisture content, vegetation effects) may not be fully captured. Regarding data efficiency, the analysis of point cloud density shows that MLS-based approaches can support accurate forest inventories even at reduced sampling levels, although sufficient point density remains critical to avoid error inflation, particularly for DBH estimation.

Future research will address the current limitations by validating the system in more challenging forest conditions (e.g., thick fallen leaves, long and hard shrubs, rolling stones, deep pits, landslides, water currents, moist soil, sudden appearance of animals), and by incorporating real-world experimental data to refine the soil–wheel interaction models and reduce the simulation-to-reality gap. Additional efforts will focus on tighter integration between perception and traversability-aware planning, cooperation with UAVs, and the development of collaborative multi-robot strategies for long-term and large-scale monitoring. Further studies will also extend the analysis of point cloud density to additional forest attributes, such as canopy volume and biomass, using MLS data collected by autonomous robotic platforms. Overall, this work demonstrates how the integration of autonomous mobile robotics, advanced perception, and learning-based navigation can significantly enhance forest surveying capabilities, paving the way toward scalable, autonomous, and sustainable forest management and long-term environmental monitoring.

Author Contributions: Conceptualization, G.C., A.D.L., L.S., and S.S.; methodology, G.C., A.D.L., L.S., S.S., and E.M.; software, E.M., D.T.F., S.C., and K.B.; validation, E.M., L.S., D.T.F., S.C., and K.B.; writing—original draft preparation, G.C., L.S., E.M., and S.S.; writing—review and editing, G.C., A.G., L.S., E.M., A.D.L., and S.S.; supervision, G.C., L.S., S.S., E.M., A.G., G.A., and F.M.; project administration, G.C., A.D.L., L.S., S.S., A.G., G.A., and F.M.; funding acquisition, G.C., A.D.L., L.S., S.S., A.G., G.A., and F.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was developed within the Laboratory for Big Data, IoT, Cyber Security (LABIC) funded by Friuli Venezia Giulia region (Italy), and the Laboratory for Artificial Intelligence for Human-Robot Collaboration (AI4HRC) funded by Fondazione Friuli (Italy). This study was carried out within the PRIN 2022 project “An Artificial Intelligence Approach for Forestry Robotics in Environment Survey and Inspection (AI4FOREST)” funded by the European Union Next Generation EU (National Recovery and Resilience Plan (PNRR), Mission 4, Component 2, Investment 1.1, CUP G53D23002880001, project code 2022LP4ASR), within the Agritech National Research Center funded by the European Union Next Generation EU (National Recovery and Resilience Plan, Mission 4, Component 2, Investment 1.4, D.D. 1032 17/06/2022, CN00000022, CUP G23C22001100007), and within the research activities of the consortium iNEST (Interconnected Nord-Est Innovation Ecosystem) funded by the European Union Next Generation EU (National Recovery and Resilience Plan, Mission 4, Component 2, Investment 1.5, D.D. 1058 23/06/2022, ECS_00000043, CUP G23C22001130006).

Data Availability Statement: Data will be made available on request.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

2D	Two-dimensional
3D	Three-dimensional
AI	Artificial Intelligence
AI4FOREST	An Artificial Intelligence Approach for Forestry Robotics in Environment Survey and Inspection
AI4HRC	Artificial Intelligence for Human-Robot Collaboration
CM	Control Module
CNN	Convolutional Neural Network
CSF	Cloth Simulation Filter
DBH	Diameter at Breast Height
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DT	Digital Twin
DTM	Digital Terrain Model
ENU	East North Up
EU	European Union
GNSS	Global Navigation Satellite Systems
GPU	Graphics Processing Unit
ID	Identifier
IMU	Inertial Measurement Unit
IoT	Internet of Things
LABIC	Laboratory for Big Data, IoT, Cyber Security
LiDAR	Light Detection and Ranging
LIO-SAM	LiDAR Inertial Odometry via Smoothing and Mapping
ML	Machine Learning
MLS	Mobile Laser Scanning
PID	Proportional Integral Derivative
PNRR	National Recovery and Resilience Plan
PRIN	Research Projects of Significant National Interest
RGB	Red, Green, Blue
RMSE	Root Mean Square Error
ROS	Robot Operating System
SDG	Sustainable Development Goal
SLAM	Simultaneous Localization and Mapping
SM	Surrogate Model
TEB	Timed Elastic Band
TH	Tree Height
UAV	Unmanned Aerial Vehicle

References

1. FAO. *Global Forest Resources Assessment, 2020: Main report*; Food and Agriculture Organization of the United Nations: Rome, Italy, 2020.
2. Aguiar, A.S.; Dos Santos, F.N.; Cunha, J.B.; Sobreira, H.; Sousa, A.J. Localization and mapping for robots in agriculture and forestry: A survey. *Robotics* **2020**, *9*, 97. [\[CrossRef\]](#)
3. Dainelli, R.; Toscano, P.; Di Gennaro, S.F.; Matese, A. Recent advances in Unmanned Aerial Vehicles forest remote sensing—A systematic review. Part II: Research applications. *Forests* **2021**, *12*, 397. [\[CrossRef\]](#)
4. Ahola, J.M.; Heikkilä, T.; Raitila, J.; Sipola, T.; Tenhunen, J. Estimation of breast height diameter and trunk curvature with linear and single-photon LiDARs. *Ann. For. Sci.* **2021**, *78*, 79. [\[CrossRef\]](#)
5. Tremblay, J.F.; Béland, M.; Gagnon, R.; Pomerleau, F.; Giguère, P. Automatic three-dimensional mapping for tree diameter measurements in inventory operations. *J. Field Robot.* **2020**, *37*, 1328–1346. [\[CrossRef\]](#)
6. Yazdi, H.; Boey, K.Z.; Rötzer, T.; Petzold, F.; Shu, Q.; Ludwig, F. Automated classification of tree species using graph structure data and neural networks. *Ecol. Inform.* **2024**, *84*, 102874. [\[CrossRef\]](#)

7. Sanchez-Guzman, G.; Velasquez, W.; Alvarez-Alvarado, M.S. Modeling a simulated forest to get burning times of tree species using a digital twin. In Proceedings of the 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 26–29 January 2022; IEEE: New York, NY, USA, 2022; pp. 0639–0643.
8. Maset, E.; Scalera, L.; Beinat, A.; Visintini, D.; Gasparetto, A. Performance investigation and repeatability assessment of a mobile robotic system for 3D mapping. *Robotics* **2022**, *11*, 54. [[CrossRef](#)]
9. Ziparo, V.A.; Zaratti, M.; Grisetti, G.; Bonanni, T.M.; Serafin, J.; Di Cicco, M.; Proesmans, M.; Van Gool, L.; Vysotska, O.; Bogoslavskyi, I.; et al. Exploration and mapping of catacombs with mobile robots. In Proceedings of the 2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Linköping, Sweden, 21–26 October 2013; IEEE: New York, NY, USA, 2013; pp. 1–2.
10. Stavridis, S.; Droukas, L.; Doulgeri, Z.; Papageorgiou, D.; Dimeas, F.; Soriano, Á.; Molina, S.; Deiri, S.A.; Hutchinson, M.; Pulido-Fentanes, J.; et al. Robotic Grape Inspection and Selective Harvesting in Vineyards: A Multisensory Robotic System With Advanced Cognitive Capabilities. *IEEE Robot. Autom. Mag.* **2024**, *32*, 51–63. [[CrossRef](#)]
11. Mammarella, M.; Comba, L.; Biglia, A.; Dabbene, F.; Gay, P. Cooperation of unmanned systems for agricultural applications: A theoretical framework. *Biosyst. Eng.* **2022**, *223*, 61–80. [[CrossRef](#)]
12. Tiozzo Fasiolo, D.; Scalera, L.; Maset, E.; Gasparetto, A. Recent Trends in Mobile Robotics for 3D Mapping in Agriculture. In *Advances in Service and Industrial Robotics. RAAD 2022*; Müller, A., Brandstötter, M., Eds.; Mechanisms and Machine Science; Springer: Cham, Switzerland, 2022; Volume 120.
13. Gasparino, M.V.; Sivakumar, A.N.; Liu, Y.; Velasquez, A.E.; Higuti, V.A.; Rogers, J.; Tran, H.; Chowdhary, G. Wayfast: Navigation with predictive traversability in the field. *IEEE Robot. Autom. Lett.* **2022**, *7*, 10651–10658. [[CrossRef](#)]
14. Oliveira, L.F.P.; Moreira, A.P.; Silva, M.F. Advances in forest robotics: A state-of-the-art survey. *Robotics* **2021**, *10*, 53. [[CrossRef](#)]
15. Pierzchała, M.; Giguère, P.; Astrup, R. Mapping forests using an unmanned ground vehicle with 3D LiDAR and graph-SLAM. *Comput. Electron. Agric.* **2018**, *145*, 217–225. [[CrossRef](#)]
16. Sheng, Y.; Zhao, Q.; Wang, X.; Liu, Y.; Yin, X. Tree diameter at breast height extraction based on mobile laser scanning point cloud. *Forests* **2024**, *15*, 590. [[CrossRef](#)]
17. Malladi, M.V.; Guadagnino, T.; Lobefaro, L.; Mattamala, M.; Griess, H.; Schweier, J.; Chebrolu, N.; Fallon, M.; Behley, J.; Stachniss, C. Tree instance segmentation and traits estimation for forestry environments exploiting LiDAR data collected by mobile robots. In Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan, 13–17 May 2024; IEEE: New York, NY, USA, 2024; pp. 17933–17940.
18. Freißmuth, L.; Mattamala, M.; Chebrolu, N.; Schaefer, S.; Leutenegger, S.; Fallon, M. Online tree reconstruction and forest inventory on a mobile robotic system. In Proceedings of the 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Abu Dhabi, United Arab Emirates, 14–18 October 2024; IEEE: New York, NY, USA, 2024; pp. 11765–11772.
19. Zhang, X.; Liu, Y.; Liu, J.; Chen, X.; Xu, R.; Ma, W.; Zhang, Z.; Fu, S. An autonomous navigation system with a trajectory prediction-based decision mechanism for rubber forest navigation. *Sci. Rep.* **2024**, *14*, 29495. [[CrossRef](#)] [[PubMed](#)]
20. Ni, J.; Chen, Y.; Tang, G.; Cao, W.; Yang, S.X. An Integration Model of Blind Spot Estimation and Traversable Area Detection for Indoor Robots. *IEEE Sens. J.* **2025**, *25*, 17850–17866. [[CrossRef](#)]
21. Ali, M.; Jardali, H.; Roy, N.; Liu, L. Autonomous navigation, mapping and exploration with gaussian processes. In Proceedings of the Robotics: Science and Systems (RSS), Daegu, Republic of Korea, 14 July 2023.
22. Fahnestock, E.; Fuentes, E.; Prentice, S.; Vasilopoulos, V.; Osteen, P.R.; Howard, T.; Roy, N. Far-Field Image-Based Traversability Mapping for A Priori Unknown Natural Environments. *IEEE Robot. Autom. Lett.* **2025**, *10*, 6039–6046. [[CrossRef](#)]
23. Wolk, K.; Tatara, M.S. A review of semantic segmentation and instance segmentation techniques in forestry using LiDAR and imagery data. *Electronics* **2024**, *13*, 4139. [[CrossRef](#)]
24. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5100–5109.
25. Kim, D.H.; Ko, C.U.; Kim, D.G.; Kang, J.T.; Park, J.M.; Cho, H.J. Automated segmentation of individual tree structures using deep learning over LiDAR point cloud data. *Forests* **2023**, *14*, 1159. [[CrossRef](#)]
26. Krisanski, S.; Taskhiri, M.S.; Gonzalez Aracil, S.; Herries, D.; Muneri, A.; Gurung, M.B.; Montgomery, J.; Turner, P. Forest structural complexity tool—an open source, fully-automated tool for measuring forest point clouds. *Remote Sens.* **2021**, *13*, 4677. [[CrossRef](#)]
27. Xiang, B.; Wielgosz, M.; Kontogianni, T.; Peters, T.; Puliti, S.; Astrup, R.; Schindler, K. Automated forest inventory: Analysis of high-density airborne LiDAR point clouds with 3D deep learning. *Remote Sens. Environ.* **2024**, *305*, 114078. [[CrossRef](#)]
28. Henrich, J.; van Delden, J.; Seidel, D.; Kneib, T.; Ecker, A.S. TreeLearn: A deep learning method for segmenting individual trees from ground-based LiDAR forest point clouds. *Ecol. Inform.* **2024**, *84*, 102888. [[CrossRef](#)]
29. da Silva, D.Q.; Dos Santos, F.N.; Sousa, A.J.; Filipe, V. Visible and thermal image-based trunk detection with deep learning for forestry mobile robotics. *J. Imaging* **2021**, *7*, 176. [[CrossRef](#)] [[PubMed](#)]

30. da Silva, D.Q.; dos Santos, F.N.; Filipe, V.; Sousa, A.J.; Oliveira, P.M. Edge AI-based tree trunk detection for forestry monitoring robotics. *Robotics* **2022**, *11*, 136. [\[CrossRef\]](#)
31. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
32. Cai, Z.; Vasconcelos, N. Cascade R-CNN: High quality object detection and instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 1483–1498. [\[CrossRef\]](#) [\[PubMed\]](#)
33. Grondin, V.; Fortin, J.M.; Pomerleau, F.; Giguère, P. Tree detection and diameter estimation based on deep learning. *Forestry* **2023**, *96*, 264–276. [\[CrossRef\]](#)
34. Zhou, S.; Xi, J.; McDaniel, M.W.; Nishihata, T.; Saleses, P.; Iagnemma, K. Self-supervised learning to visually detect terrain surfaces for autonomous robots operating in forested terrain. *J. Field Robot.* **2012**, *29*, 277–297. [\[CrossRef\]](#)
35. Scalera, L.; Tiozzo Fasiolo, D.; Maset, E.; Carabin, G.; Seriani, S.; De Lorenzo, A.; Alberti, G.; Gasparetto, A. Mobile Robotics for Forest Monitoring and Mapping Within the AI4FOREST Project. In *Mechanical Engineering Solutions: Design, Simulation, Testing, Manufacturing*. MES 2025; Parikyan, T., Sargsyan, Y., Ceccarelli, M., Eds.; Mechanisms and Machine Science; Springer: Cham, Switzerland, 2026; Volume 191.
36. Shan, T.; Englot, B.; Meyers, D.; Wang, W.; Ratti, C.; Rus, D. LIO-SAM: Tightly-coupled LiDAR Inertial Odometry via Smoothing and Mapping. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; IEEE: New York, NY, USA, 2020; pp. 5135–5142.
37. Cottiga, S.; Bonin, L.; Giberna, M.; Caruso, M.; Görner, M.; Carabin, G.; Scalera, L.; De Lorenzo, A.; Seriani, S. Leveraging Machine Learning for Terrain Traversability in Mobile Robotics. In *Mechanism Design for Robotics*. MEDER 2024; Lovasz, E.C., Ceccarelli, M., Ciupe, V., Eds.; Mechanisms and Machine Science; Springer: Cham, Switzerland, 2024; Volume 166.
38. Marder-Eppstein, E.; Chitta, S. Carrot Planner. 2018. Available online: https://wiki.ros.org/carrot_planner (accessed on 20 December 2025).
39. Rösmann, C.; Hoffmann, F.; Bertram, T. Integrated online trajectory planning and optimization in distinctive topologies. *Robot. Auton. Syst.* **2017**, *88*, 142–153. [\[CrossRef\]](#)
40. Schubert, E.; Sander, J.; Ester, M.; Kriegel, H.P.; Xu, X. DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Trans. Database Syst. (TODS)* **2017**, *42*, 1–21. [\[CrossRef\]](#)
41. Tiozzo Fasiolo, D.; Maset, E.; Scalera, L.; Macaulay, S.; Gasparetto, A.; Fusiello, A. Combining LiDAR SLAM and deep learning-based people detection for autonomous indoor mapping in a crowded environment. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2022**, *43*, 447–452. [\[CrossRef\]](#)
42. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
43. Koenig, N.; Howard, A. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), Sendai, Japan, 28 September–2 October 2004; Volume 3, pp. 2149–2154. [\[CrossRef\]](#)
44. Pavlov, C.; Johnson, A.M. A terramechanics model for high slip angle and skid with prediction of wheel-soil interaction geometry. *J. Terramechanics* **2024**, *111*, 9–19. [\[CrossRef\]](#)
45. Zhou, R.; Feng, W.; Ding, L.; Yang, H.; Gao, H.; Liu, G.; Deng, Z. MarsSim: A High-Fidelity Physical and Visual Simulation for Mars Rovers. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *59*, 1879–1892. [\[CrossRef\]](#)
46. Caruso, M.; Bregant, L.; Gallina, P.; Seriani, S. Design and multi-body dynamic analysis of the Archimede space exploration rover. *Acta Astronaut.* **2022**, *194*, 229–241. [\[CrossRef\]](#)
47. Caruso, M.; Giberna, M.; Görner, M.; Gallina, P.; Seriani, S. The Archimede Rover: A Comparison between Simulations and Experiments. *Robotics* **2023**, *12*, 125. [\[CrossRef\]](#)
48. Kankare, V.; Puttonen, E.; Holopainen, M.; Hyypä, J. The Effect of TLS Point Cloud Sampling on Tree Detection and Diameter Measurement Accuracy. *Remote Sens. Lett.* **2016**, *7*, 495–502. [\[CrossRef\]](#)
49. Ruiz, L.; Hermosilla, T.; Mauro, F.; Godino, M. Analysis of the Influence of Plot Size and LiDAR Density on Forest Structure Attribute Estimates. *Forests* **2014**, *5*, 936–951. [\[CrossRef\]](#)
50. Laino, D.; Cabo, C.; Prendes, C.; Janvier, R.; Ordóñez, C.; Nikonovas, T.; Doerr, S.; Santin, C. 3DFin: A Software for Automated 3D Forest Inventories from Terrestrial Point Clouds. *For. Int. J. For. Res.* **2024**, *97*, 479–496. [\[CrossRef\]](#)
51. Muhojoki, J.; Tavi, D.; Hyypä, E.; Lehtomäki, M.; Faitli, T.; Kaartinen, H.; Kukko, A.; Hakala, T.; Hyypä, J. Benchmarking Under- and Above-Canopy Laser Scanning Solutions for Deriving Stem Curve and Volume in Easy and Difficult Boreal Forest Conditions. *Remote Sens.* **2024**, *16*, 1721. [\[CrossRef\]](#)
52. 3DFin (Plugin)—CloudCompareWiki. Available online: https://www.cloudcompare.org/doc/wiki/index.php/3DFin_%28plugin%29 (accessed on 15 December 2025).

53. Cabo, C.; Ordóñez, C.; López-Sánchez, C.A.; Armesto, J. Automatic Dendrometry: Tree Detection, Tree Height and Diameter Estimation Using Terrestrial Laser Scanning. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *69*, 164–174. [[CrossRef](#)]
54. Zhang, W.; Qi, J.; Wan, P.; Wang, H.; Xie, D.; Wang, X.; Yan, G. An Easy-to-Use Airborne LiDAR Data Filtering Method Based on Cloth Simulation. *Remote Sens.* **2016**, *8*, 501. [[CrossRef](#)]
55. Khan, N.A.; Carabin, G.; Mazzetto, F. Mobile Laser Scanning in Forest Inventories: Testing the Impact of Point Cloud Density on Tree Parameter Estimation. *Sensors* **2025**, *25*, 5798. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.