

Leveraging Behavior Trees for Hybrid Autonomous Navigation in Seasonal Agricultural Environments

Juan Francisco Rascón, Pau Reverté, Xavier Ruiz, Mateus S. Moura, Daniel Serrano and Carlos Rizzo
Eurecat, Centre Tecnològic de Catalunya, Robotics and Automation Unit. Cerdanyola del Vallès, Spain
{juan.rascon, pau.reverte, xavi.ruiz, mateus.sanches, daniel.serrano, carlos.rizzo}@eurecat.org

Abstract—Autonomous navigation is essential for the successful integration of mobile robots in agricultural operations. In structured fields, where permanent crops are usually disposed in row patterns, perception-based navigation is typically used for achieving safe and efficient in-row operation, whereas map-based navigation and other techniques are applied for transitioning between rows and from/to the robot base station. However, execution and coordination of different strategies has been mostly achieved using finite state machines or rule-based implementations, limiting orchestration of complex behaviors and scalability. This work presents a modular reasoning architecture that leverages behavior trees and a topological representation of the environment for deploying agricultural robots, switching between operation modes (perception or map-based) according to their topological state and goal, and embedding recovery behaviors in the event of failure. The system has been validated with different robotic platforms (mobile robot and retrofitted tractor) and large-scale pilots (apple orchards and table grape vineyards), resulting in successful autonomous spraying demonstrations.

Index Terms—mobile robot, autonomous navigation, behavior tree, path planning, agriculture

I. INTRODUCTION

Precision farming has become of great interest to farmers, researchers and policy makers, as optimizing crop yield while reducing resource usage and environmental impact are desired to the agri-food sector and the green deal. Towards such, automation of base crop management operations with mobile robots is ideal, given their capacity to execute heavy, repetitive and hazardous tasks with accuracy and efficiency [1]. In open fields, GNSS-guided tractors and agri-robots are at a mature technological level, embedding automated planning and collision avoidance functions while executing operations such as spraying, weeding and harvesting. However, for perennial crops such as orchards and vineyards, their rough, dynamic and seasonal nature require advanced strategies for robot localization, planning and navigation [2].

Most navigation techniques in structured fields leverage their patterns with crop row following methods [2], as these are more robust than traditional waypoint following or grid map planning. These better suit navigation outside crop row driveways (e.g., changing rows, moving to warehouse), as the environment is subject to fewer traversability restrictions. Coordinating between those different approaches therefore require a mission planner that triggers navigation modes and

This work has been developed under the Robs4Crops European project and received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no. 101016807.

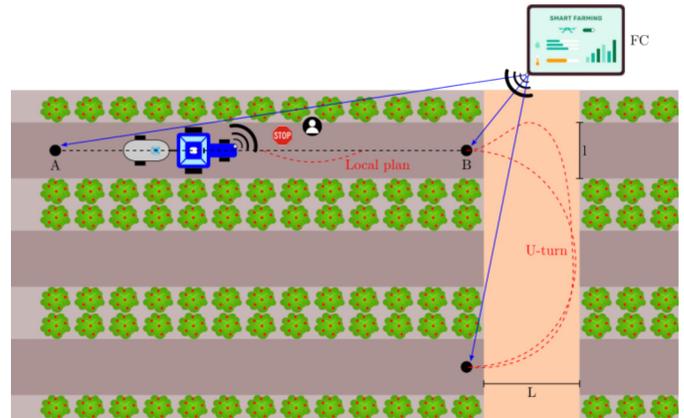


Fig. 1: Illustration of the proposed system, with two autonomous navigation modes (in-row, inter-row) highlighted.

transitions between them. Besides, the complex and safety-critical operations conducted with heavy machinery require orchestrating diverse behaviors and gracefully handling failure events with recovery actions, where standard state machine or rule-based navigation are limited. In this sense, behavior trees (BT) have gained attention in the robotics community [3] as a way to naturally express complex plans in a modular, scalable fashion with robustness to failure, albeit not sufficiently explored in agricultural robotics.

For such reason, we present a novel autonomous navigation architecture (illustrated in Figure 1) that implements behavior tree-based navigation modes, reasoning upon a topological representation of perennial crop fields, and applicable to different operations and field types across the crop cycle. To validate our design choices, we performed experiments in large scale pilot (LSP) fields corresponding to target scenarios, with different robots and across seasons, evidencing a safe, robust and efficient navigation architecture for agricultural operations.

Thus, our contributions are the following:

- A crop row estimation pipeline based on point cloud clustering and linear regression for tracking parallel crop rows and calculating the central driveway for row following.
- Navigation behavior trees with embedded recovery behaviors for safe and efficient row following, transitioning between rows, and navigation outside crop parcels.
- A mission commander integrated with a GUI-based farming controller, that orchestrates the different behavior

trees for providing full in-field autonomous navigation.

The remainder of this paper is structured as follows. Section II provides a brief state of the art review on mobile robot navigation in structured fields, including hybrid approaches. Section III presents the proposed navigation solution, detailing each component. Section IV explains the materials and methods used for system evaluation, including the robotic platforms and large scale pilot fields used. Section V presents the results achieved and provides a brief discussion on them. Finally, Section VI gives concluding remarks and future directions.

II. RELATED WORK

Standard autonomous navigation approaches in agriculture include point-to-point planning using map-based localization [4], [5], waypoint-based GNSS positioning [6], rule-based heuristics [7] and finite state machines [8], [9]. Topological and hybrid field representations have been recently explored [10]–[12] as a way to embed different navigation strategies, typically including initialization, row following, U-turning, and go-to-goal methods. In [13], a task-driven planning and execution system was developed, where tasks are represented by state diagrams and managed by a mission planner.

Behavior trees have been adopted for autonomous navigation with mobile robots, mainly within the standard de-facto ROS-based `nav2` system [14] which provides diverse planners and controllers for generic applications, as well as recovery behaviors. Domain-specific applications are found in the literature for search and rescue, service robotics, logistics, and autonomous driving [3], whereas behavior-based solutions for accomplishing agricultural tasks exist in the literature [15], [16], but no specific implementation with behavior trees has been identified.

With respect to reactive navigation strategies in crop row fields, row following techniques are the main subject of research, based either on point cloud clustering, Hough transform, or vanishing point estimation from images [17], [18]. Detecting parallel rows [9], [19] adds robustness by averaging multiple estimations or with a patterned Hough transform. Conversely, transitioning between rows is a less studied problem, addressed by [19], [20].

III. SYSTEM ARCHITECTURE

The proposed autonomous navigation system consists of a perception module that estimates crop row lines for reactive navigation, a localization module to estimate the robot global pose, a hybrid planner that contains multiple navigation modes implemented as behavior trees, and a mission commander that orchestrates such modes over a topological map, integrated with an external farming controller GUI. The system has been implemented in C++ within the ROS¹ 2 robotics middleware.

A. Crop Row Perception

The perception pipeline for reactive navigation (Figure 2) consists in a crop hallway detector that leverages the structure

of crop rows based on sensor data, estimating parallel line segments that best represent the crop row driveway, coupled with a row estimator, which derives the central reference path (with respect to the robot frame) from the estimated crop line segments. Such path becomes the reference for the in-row navigation mode (Section III-D).

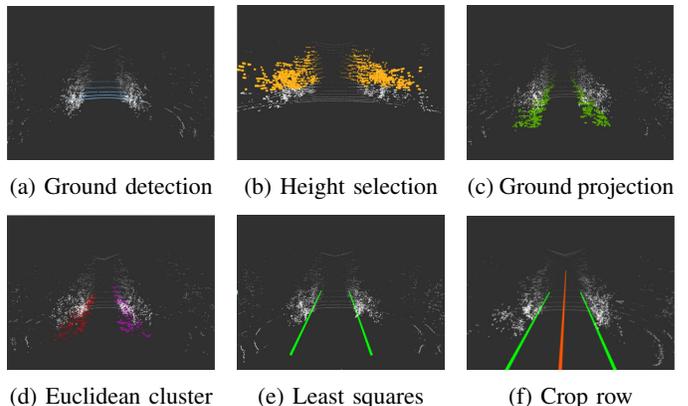


Fig. 2: Overview of the crop row perception pipeline: hallway detector (a-e), row estimator (f).

Hallway Line Detection: using 3D LIDAR information, this algorithm first segments the ground plane from the input point cloud using RANSAC [21], followed by point filtering based on a distance range $[d_{min}, d_{max}]$ to the ground plane, and point projection ($p \rightarrow p_{proj}$) onto it as:

$$p_{proj} = p - d \cdot \vec{n} \quad (1)$$

where \vec{n} is the unit vector normal to the ground plane and d is the point-to-plane distance. Data down-sampling and Euclidean cluster extraction are then applied to identify crop row clusters within the projected cloud, assuming the two most dense clusters as being the hallway “walls”. Finally, a Least Squares method is used to fit two parallel lines to these clusters, modeled as:

$$y = m \cdot x + b + i \cdot s, i \in \mathbb{Z} \text{ and } i \in \{0, 1\} \quad (2)$$

with m and b being line coefficients, s the separation between parallel lines, and i as the line index. Solving it for all points yields the set of parallel lines that best fit the clusters.

Crop Row Estimation: based on the detected hallway line segments, these are filtered by distance and orientation thresholds (based on the previous estimate) and then classified as left or right according to their position relative to the robot. The distance between both lines is checked against the crop row width within a given tolerance and the central line is then calculated as the midpoint between them.

B. Robot Localization

In orchards and vineyards, GNSS signals are not always available or accurate due to blockage or multi-path effects by crops or structural elements, and traditional wheel odometry presents considerable drift in uneven terrain. For this reason,

¹<https://www.ros.org/>

we implemented a filtering approach for in-field localization, fusing multiple sensors and estimators with a dual extended Kalman filter (EKF), providing both position and orientation robust to uncertainty or failure in any of the sources. Figure 3 presents the diagram of the localization subsystem, where the local and global EKF modules and their input components are shown.

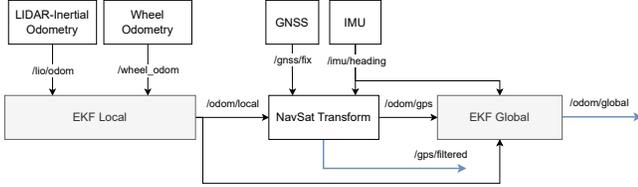


Fig. 3: Localization subsystem diagram.

The local EKF module relies on a fast and accurate lidar-inertial odometry (LIO), based on the FAST-LIO2 [22] method and complemented with wheel odometry provided by the robotic platforms, for estimating short-term motion. Subsequently, the global EKF module fuses the continuous, high-frequency information from the short-term odometry with low-frequency, absolute position coordinates, as well as heading estimates from a moving baseline setup using multiple GNSS receivers. At last, given a set of origin coordinates, conversions between geodetic and cartesian systems are managed.

C. Topological Environment Representation

The proposed system relies on a topological field description to perform hybrid planning and autonomous navigation, illustrated in Figure 4 and composed of the following entities:

Row parcel: a rectangular zone representing the region delimited by consecutive crop rows, containing information about its dimensions (width, length) and entry/exit points.

Free parcel: a polygonal zone representing an unoccupied region around the crops, where robots can navigate freely.

Points of Interest: reference positions inside free parcels where the robot can be deployed in a mission (e.g., warehouses, recharge stations, docks, as well as entry/exit points from row parcels).



Fig. 4: Sample field topological representation with delimited parcels and interest points.

Both parcels and interest points are user-provided and defined by (sets of) global coordinates. Once defined, the topological map is fed to the commander (Section III-E).

D. Hybrid Behavior Tree-Based Planning and Control

The core of the proposed architecture is a hybrid planning and control approach consisting of the following autonomous navigation modes:

In-row navigation: active when the robot is located inside a row parcel. The navigation goal consists in traversing the row until its end, while keeping the robot centered along the row for navigating safely and correctly performing treatment tasks. Due to the varying nature of crops, a reactive navigation pipeline is preferred, as it adapts to the perceived crop distribution. Using the row estimator described in Section III-A, the row line relative to the robot frame is continuously updated and serves as a reference for the robot controller.

Inter-row navigation: active when the robot transitions from one row parcel to another. The exit and entry points of source and target row parcels, respectively, are used to calculate a fixed reference trajectory in the global frame composed of Dubins curves. The involved headland width and separation between rows, as well as kinematic restrictions of the robot (e.g. minimum turning radius), are also considered for trajectory generation.

Free navigation: active when the robot navigates between two interest points in a free parcel. The SMAC [23] Hybrid-A* planner is used for computing the best path considering a cost map of the free parcel and the robot footprint and kinematics.

Figure 5 illustrates the transition between in- and inter-row navigation modes between two parcels in a real orchard, displaying the trajectories for row following and transition.

The navigation modes are implemented as behavior trees using the BehaviorTree.CPP² library, a framework to generate custom action, condition, control and decorator nodes and composing them into the tree’s hierarchical structure, allowing complex behaviors through reusable components. The execution flow of a tree starts at the root node and propagates based on the control flow and the execution result of nodes (RUNNING, SUCCESS or FAILURE), permitting the inclusion of recovery behaviors for when a given action is not successful (e.g., when an obstacle blocks the driveway, attempt to re-calculate the path towards a goal). Each navigation mode has its corresponding tree, meaning that the mode will be active until the tree is terminated. Figure 6 shows the BT implemented for the in-row navigation mode, as an example.

All navigation modes rely on a standard Pure Pursuit [24] controller for path tracking, for its compatibility with car-type (Ackermann) kinematics, dynamically adjusting the steering direction to follow a look-ahead target point over the reference path, by setting a fixed distance ahead of the vehicle.

E. Mission Commander and Integration with GUI

The mission commander is responsible for managing and executing a sequence of navigation sub-missions, each of

²<https://www.behaviortree.dev/>

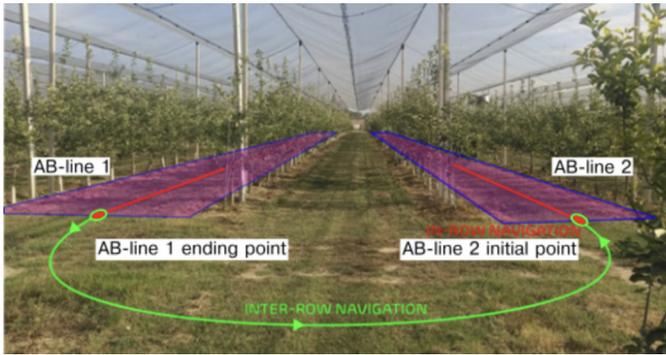


Fig. 5: Navigation modes transition example.

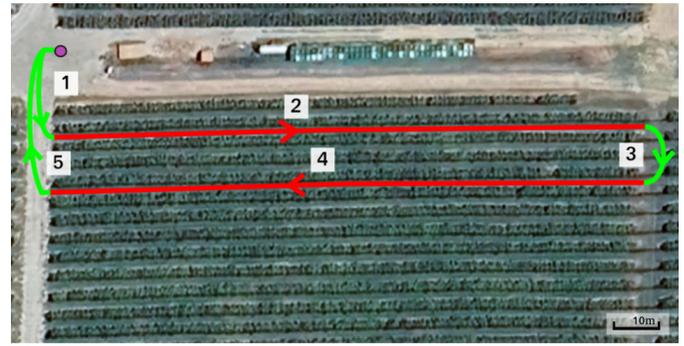


Fig. 7: Sample two-row mission sequence generated by the commander, decomposed in the following navigation modes: free (1, 5), in-row (2, 4) and inter-row (3).

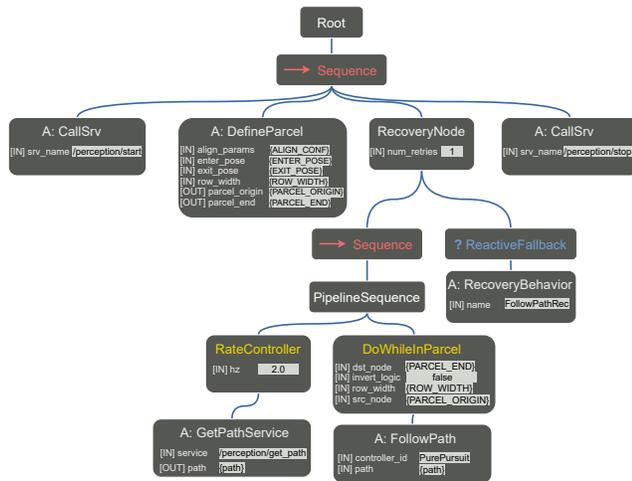


Fig. 6: In-row navigation behavior tree.

which consists in running a navigation mode BT with mode-dependent execution parameters (e.g., target row or goal). For instance, a mission of treating two row parcels and returning to the warehouse (depicted in Figure 7) can be decomposed into a free navigation to the first row entry point, followed by in-row navigation, maneuvering to the second row, in-row navigation and then free navigation back to the warehouse.

Once this sequence is generated, the commander activates each BT by periodically running (*ticking*) it to ensure progression, as most of the actions are asynchronous. Execution begins with the first BT, performing its tasks to achieve its goal. Upon completion, the commander executes the next tree. This iterative ticking continues, with the commander monitoring each BT status until the mission completes when the final tree is executed. In the event of BT failure where no recovery actions were successful, the mission is aborted.

The mission commander is integrated with an external farming controller (Figure 8), a cloud-based GUI that generates agricultural missions and provides real-time feedback with interfaces for starting, pausing or cancelling missions. Users can select tasks and optimization policies, and the farming controller sends missions to the robotic platform that are received and orchestrated by the commander.



Fig. 8: Farming Controller GUI.

IV. EXPERIMENTAL METHODOLOGY

A. Large Scale Pilots

For evaluation of the proposed navigation system, two of the project LSP fields were chosen to perform validation experiments and public demonstrations: an apple orchard in the region of Girona, Spain (Figure 9, top) and a table grape vineyard in Kiato, Greece (Figure 9, bottom). Both scenarios served for testing and trials of automated navigation and spraying remotely commanded through the farming controller:

Field 1: Apple Orchard

- **Coordinates:** Latitude 42.1628, Longitude 3.0930
- **Crop Type:** Apple Orchard
- **Row Width:** 3.8 meters
- **Row Length:** 130 meters
- **Area:** 3.8 hectares

Field 2: Vineyard

- **Coordinates:** Latitude 37.9437, Longitude 22.7720
- **Crop Type:** Vineyard (Table Grapes)
- **Row Width:** 2.5 meters
- **Row Length:** 65 meters
- **Area:** 0.5 hectares

To evaluate the effectiveness of the row perception methods for reactive navigation, the system was tested in the Girona orchard pilot across different growth stages. Tests were conducted in early spring (no foliage) and late summer



Fig. 9: Large scale pilot fields considered for the experiments: apple orchard (top), table grape vineyard (bottom).



Fig. 10: Evaluation platforms: NH T4.110f retrofitted tractor (left), AGC CAROB agri-robot (right).

(full foliage) to assess robustness under varying environmental conditions, while the vineyard trials in Kiato occurred only during summer. The positioning ground truth used to evaluate the perception of the reactive path during in-row navigation is provided by the dual extended Kalman filter (EKF) which effectively compensates for GNSS signal shortages.

B. Evaluation Platforms

To demonstrate adaptability to different robotic platforms, we deployed the proposed system for real-time execution in a retrofitted tractor and a mobile agri-robot (Figure 10).

Retrofitted Tractor: For the apple orchard pilot, a New Holland (NH) T4.110f autonomous tractor with car-type kinematics was used. The tractor was retrofitted by undergoing electronic and mechanic modifications on gas, steering and brake systems to allow remote control. The sensors installed for data acquisition are a RSHelios-16P 3D LIDAR, an OAKD-LR Stereo camera, a UM7 IMU and an AGCBox dual GNSS receiver and antennas.

Mobile Agri-Robot: For the vineyard pilot, an AGreen-culture (AGC) CAROB tracked robot was employed, whose

dimensions and differential kinematics simplify planning and navigation stages by providing higher mobility. The same sensor setup as of the retrofitted tractor was used.

V. RESULTS AND DISCUSSION

Following the autonomous spraying trials in the orchard and vineyard pilots, Figure 11 shows the resulting trajectories for each of the missions accomplished, along with point cloud maps of the fields generated by the LIO module. In both cases, the system was capable of freely navigating towards a row entry, followed by reactively crossing two row parcels while executing transitions between them, validating the hybrid planning and orchestration approach. During the maneuvering, the robots were capable of exiting/entering the row parcels aligned with the principal direction since minimum exit/entry distances from the parcels were preset, ensuring a smooth transition between inter-row and in-row modes. Eventual faults (obstacle detection, spraying system failure) were managed by the BT recovery behaviors, which in this case would interrupt the navigation until the conditions were solved, with no missions aborted due to BT failure.

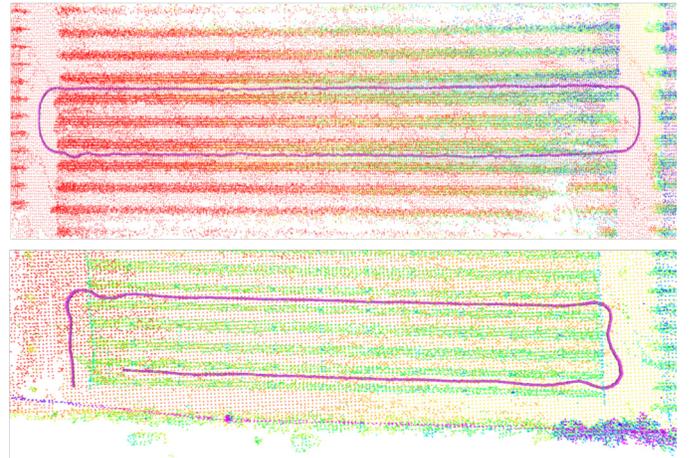


Fig. 11: Autonomous mission trajectories in apple orchard (top) and vineyard (bottom).

With respect to the perception module, Figure 12 presents the positioning error of the estimated row line relative to the equivalent commanded paths (considering a line segment over the interest points) for all considered scenarios. Table I summarizes this comparison, detailing maximum error, mean absolute error (MAE), and standard deviation for all cases. The field frondosity affects the reactive trajectory accuracy, as when vegetation is minimal or absent, the resulting path aligns closely with the theoretical row central line, while in vegetation-rich environments, the irregular shapes and obstacles introduced by foliage leads to noisy estimates and deviations between the reactive path and the reference line. Despite the differences in both cases, overall, the system operates accurately in both scenarios. No significant differences were found for different field types in similar vegetation conditions.

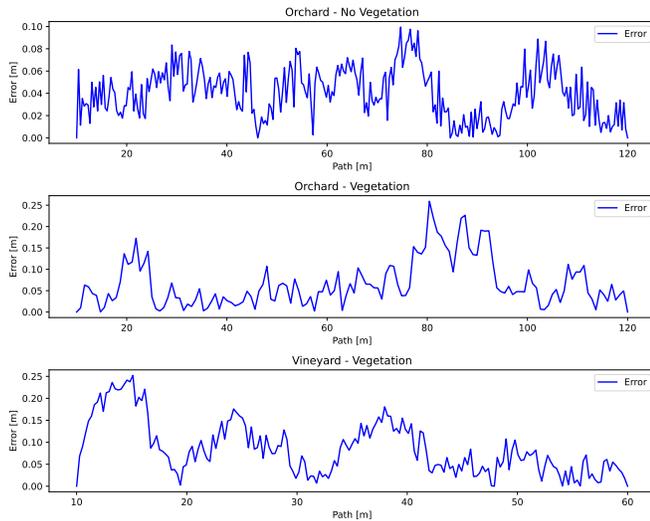


Fig. 12: Position error of the reactive path with respect to the commanded in orchard non-vegetation (top) and vegetation (middle) and vineyard in vegetation conditions (bottom).

TABLE I: Error comparison between commanded and reactive paths for different field vegetation conditions and crops.

Field conditions	Max. error [m]	MAE [m]	Std. dev. [m]
Orchard - no vegetation	0.0996	0.0406	0.0225
Orchard - with vegetation	0.2592	0.0669	0.0552
Vineyard - with vegetation	0.2520	0.0739	0.0607

Regarding the control performance, Figure 13 displays results on lateral distance to both crop sides during reactive navigation for all considered scenarios. It is noted that the average difference between distances was less than 0.1m for the non-vegetated case, and 0.2m for the vegetated one, indicating that the Pure Pursuit-based controller was able to track the central row lines accurately.

TABLE II: Distance metrics on crop lateral distance for the considered scenarios.

Field conditions	Sum [m]	MAE [m]	Std. dev. [m]
Orchard - no vegetation	3.354	0.1472	0.1439
Orchard - with vegetation	3.0966	0.2163	0.1853
Vineyard - with vegetation	2.5132	0.1275	0.1494

VI. CONCLUSION

This work presented an autonomous navigation system for agricultural robots with a focus on perennial, row-based crops such as orchards and vineyards, although in principle it could be applicable to other fields with crop row patterns such as arable fields. By orchestrating behavior trees over a topological representation of the field, the system can switch between different navigation modes while handling complex behaviors,

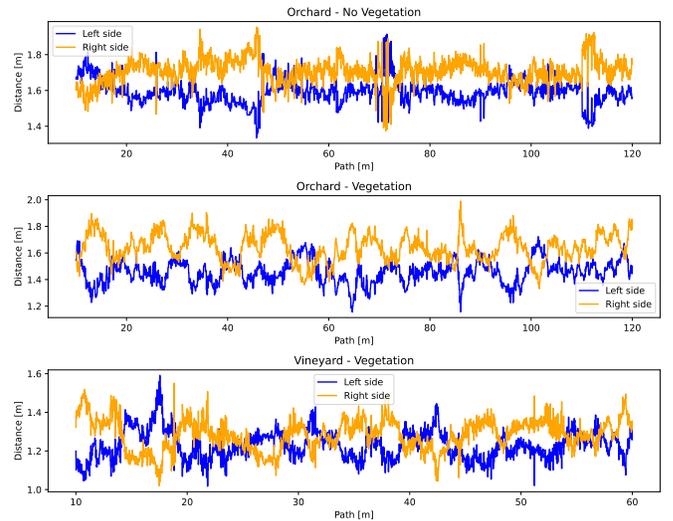


Fig. 13: Comparison of lateral distance from robot to crop sides in orchards with no vegetation (top), with vegetation (middle), and vineyard with vegetation (bottom).

such as recovery actions and implement management, within each mode. The presented architecture is therefore modular and scalable, while retaining navigation accuracy and safety, with a demonstrated potential to be integrated with agricultural farm management systems and digital twins. Future work directions include developing an automated end-of-row detection, based on detected clusters or learned features, which could facilitate the transition between in-row and inter-row navigation modes and prevent dependence on global localization systems, and extending the proposed system to other operations such as harvesting and pruning, thereby enhancing its usefulness to end-users.

ACKNOWLEDGMENT

The authors thank project partners SERRATER SL and PEGASUS COOP for providing their facilities, as well as LMS for giving access to their cloud-based farming controller and AGC for their labours in vehicle retrofitting, making possible to run the autonomous navigation and spraying experiments detailed in this work in real field conditions.

REFERENCES

- [1] A. Botta, P. Cavallone, L. Baglieri, G. Colucci, L. Tagliavini, and G. Quaglia, "A review of robots, perception, and tasks in precision agriculture," *applied mechanics*, vol. 3, no. 3, pp. 830–854, 2022.
- [2] D. T. Fasiolo, L. Scalera, E. Maset, and A. Gasparetto, "Towards autonomous mapping in agriculture: A review of supportive technologies for ground robotics," *Robotics and Autonomous Systems*, p. 104514, 2023.
- [3] M. Iovino, E. Scukins, J. Styrd, P. Ögren, and C. Smith, "A survey of behavior trees in robotics and ai," *Robotics and Autonomous Systems*, vol. 154, p. 104096, 2022.
- [4] P. Astolfi, A. Gabrielli, L. Bascetta, and M. Matteucci, "Vineyard autonomous navigation in the echord++ grape experiment," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 704–709, 2018.
- [5] S. Wang, J. Song, P. Qi, C. Yuan, H. Wu, L. Zhang, W. Liu, Y. Liu, and X. He, "Design and development of orchard autonomous navigation spray system," *Frontiers in Plant Science*, vol. 13, p. 960686, 2022.

- [6] H. Nehme, C. Aubry, T. Solatges, X. Savatier, R. Rossi, and R. Boutteau, "Lidar-based structure tracking for agricultural robots: Application to autonomous navigation in vineyards," *Journal of Intelligent & Robotic Systems*, vol. 103, no. 4, p. 61, 2021.
- [7] J. C. Andersen, O. Ravn, and N. A. Andersen, "Autonomous rule-based robot navigation in orchards," *IFAC Proceedings Volumes*, vol. 43, no. 16, pp. 43–48, 2010.
- [8] G. Riggio, C. Fantuzzi, and C. Secchi, "A low-cost navigation strategy for yield estimation in vineyards," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2200–2205.
- [9] W. Winterhalter, F. Fleckenstein, C. Dornhege, and W. Burgard, "Localization for precision navigation in agricultural fields—beyond crop row following," *Journal of Field Robotics*, vol. 38, no. 3, pp. 429–451, 2021.
- [10] L. Emmi, E. Le Flécher, V. Cadenat, and M. Devy, "A hybrid representation of the environment to improve autonomous navigation of mobile robots in agriculture," *Precision Agriculture*, vol. 22, no. 2, pp. 524–549, 2021.
- [11] L. Santos, F. N. Santos, S. Magalhães, P. Costa, and R. Reis, "Path planning approach with the extraction of topological maps from occupancy grid maps in steep slope vineyards," in *2019 IEEE international conference on autonomous robot systems and competitions (ICARSC)*. IEEE, 2019, pp. 1–7.
- [12] R. Cordova-Cardenas, L. Emmi, and P. Gonzalez-de Santos, "Enabling autonomous navigation on the farm: A mission planner for agricultural tasks," *Agriculture*, vol. 13, no. 12, p. 2181, 2023.
- [13] L. Emmi, R. Fernández, and P. Gonzalez-de Santos, "An efficient guiding manager for ground mobile robots in agriculture," *Robotics*, vol. 13, no. 1, p. 6, 2023.
- [14] S. Macenski, F. Martín, R. White, and J. G. Clavero, "The marathon 2: A navigation system," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 2718–2725.
- [15] G. Mier, J. Valente, and S. de Bruin, "Fields2cover: An open-source coverage path planning library for unmanned agricultural vehicles," *IEEE Robotics and Automation Letters*, vol. 8, no. 4, pp. 2166–2172, 2023.
- [16] D. Pour Arab, M. Spisser, and C. Essert, "3d hybrid path planning for optimized coverage of agricultural fields: A novel approach for wheeled robots," *Journal of Field Robotics*, 2024.
- [17] S. Bonadies and S. A. Gadsden, "An overview of autonomous crop row navigation strategies for unmanned ground vehicles," *Engineering in Agriculture, Environment and Food*, vol. 12, no. 1, pp. 24–31, 2019.
- [18] Y. Bai, B. Zhang, N. Xu, J. Zhou, J. Shi, and Z. Diao, "Vision-based navigation and guidance for agricultural autonomous vehicles and robots: A review," *Computers and Electronics in Agriculture*, vol. 205, p. 107584, 2023.
- [19] A. Ahmadi, L. Nardi, N. Chebrolu, and C. Stachniss, "Visual servoing-based navigation for monitoring row-crop fields," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4920–4926.
- [20] C. Peng, Z. Fei, and S. G. Vougioukas, "Depth camera based row-end detection and headland maneuvering in orchard navigation without gnss," in *2022 30th Mediterranean Conference on Control and Automation (MED)*. IEEE, 2022, pp. 538–544.
- [21] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [22] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.
- [23] J. W. Steve Macenski, Matthew Booker, "Open-source, cost-aware kinematically feasible planning for mobile and surface robotics," pp. 1–9, 2024.
- [24] R. C. Coulter, "Implementation of the pure pursuit path tracking algorithm," 1992.