

LiDAR-based Structure Tracking for Agricultural Robots: Application to Autonomous Navigation in Vineyards

Hassan Nehme* · Clément Aubry ·
Thomas Solatges · Xavier Savatier ·
Romain Rossi · Rémi Boutteau

Received: date / Accepted: date

Abstract Autonomous navigation is a key defining feature that allows agricultural robots to perform automated farming tasks. Global navigation satellite system (GNSS) technology is providing autonomous navigation solutions for current commercial robotic platforms that can achieve centimeter-level accuracy when real-time kinematic (RTK) corrections are available. However, GNSS-based solutions are expensive and require a long preparation phase where the field has to be surveyed with a GNSS rover to collect waypoints for the navigation path. An alternative navigation approach can be provided by Local perception sensors, such as LiDAR scanners, by tracking geometric features in the perceived scene. This paper presents a robust LiDAR-based solution for structure tracking along vine rows. The proposed method does not require prior field surveying, and it is insensitive to crop characteristics such as row width and spacing. Moreover, the proposed algorithm identifies and builds an online regression model of the structure. This is done by applying the Hough transform with a parameterization and search method motivated by a practical interpretation of point cloud statistics. The proposed method was tested on a commercial robotic platform in two configurations of vineyards. The experiments show that the proposed algorithm achieves consistent and accurate row tracking, which was validated against a reliable RTK-GNSS ground truth.

Keywords Agricultural robotics · Autonomous navigation · Structure tracking · LiDAR

* Corresponding author
Hassan Nehme
E-mail: h.nehme@sitia.fr

H. Nehme · C. Aubry · T. Solatges
SITIA, 7 rue de l'halbrane, 44340 Bouguenais, France. f.last@sitia.fr

R. Rossi · X. Savatier
Normandie Univ, UNIROUEN, ESIGELEC, IRSEEM, 76000 Rouen, France.
first.last@esigelec.fr

R. Boutteau
Normandie Univ, UNIROUEN, UNILEHAVRE, INSA Rouen, LITIS, 76000 Rouen, France.
first.last@univ-rouen.fr

1 Introduction

Increasing demands for food production, lack of workforce, and new environmental regulations regarding the use of chemicals are raising the interest in automated agricultural machinery as a way to improve production means. Indeed, autonomous agricultural robots offer an entirely new approach to performing automated agricultural tasks. In order to perform these tasks efficiently and safely without damaging the crops, a robot has to navigate along crop rows with a considerable level of accuracy. Global Navigation Satellite System (GNSS) technology provides robot navigation solutions that can achieve centimeter-level accuracy when Real-Time Kinematic (RTK) corrections are available. However, these solutions require a prior definition of a global path from a set of waypoints which can be collected by surveying the field using a GNSS rover. In this regard, a reliable GNSS-based solution is costly in terms of materials invested and time consumed preparing the task. This drawback has motivated a research interest in developing alternative navigation solutions based on local perception sensors.

In a local perception navigation paradigm, sensor data such as images and 3D point clouds are analyzed in order to identify and track local features in the scene. This approach has been applied to *on-road* autonomous driving by tracking local features such as lane marking [1, 19] or road boundaries [18]. In agricultural fields, crop rows represent a geometrical structure that can be exploited as a navigation clue. Crop rows vary in features across different types of cultures, ranging from small plants in market gardening to dense canopies in viticulture. Hence, feature identification can be challenging in agricultural environments due to their open and changing nature and extreme lighting conditions. In this regard, LiDAR sensors are often preferred over cameras in these situations due to their accuracy and insensitivity to lighting conditions. Common local navigation approaches require a prior 3D map of the environment [16] or a prior measurement model [13, 25] that incorporates crop-specific characteristics such as spacing width and periodicity. This makes them less suitable for versatile robots intervening on different crops with different characteristics. Hence the interest in structure tracking methods that are indifferent to specific crop characteristics and require no prior map of the environment.

This paper presents a robust LiDAR-based solution for structure tracking along vine rows. The proposed solution consists of two main steps. The first step consists of the identification of the navigation clues provided by the vine row structure as an estimation of the overall heading and the lateral position of each row. This is done by applying Hough transform on the perceived point clouds with a parameterization and a search method motivated by a practical interpretation of point cloud statistics. In the second step, frame-to-frame structure estimations are used along with the robot's motion in order to build and maintain, in real-time, a persistent regression model for each row. Finally, tracking data are provided in the form of relative localization information, i.e., lateral and angular deviations, with respect to the regression model of the row to follow. The proposed method was tested on a commercial robotic platform in two configurations of vineyards. The experiments show that the proposed algorithm achieves consistent and accurate row tracking, which was validated against a reliable RTK-GNSS ground truth. The main contributions of this paper are:

- A LiDAR-based solution to provide autonomous tracking in vineyards, which is considered as a perceptually challenging environment due to the irregular canopy shape and high vegetation density.
- The proposed solution does not require prior field surveying, and it is insensitive to crop characteristics such as row width and spacing. This reduces the effort and time invested in task preparation.
- The proposed method was implemented and tested in real conditions on production sites. The evaluation with respect to a reliable GNSS-based solution showed very high robustness and consistency, even when some vines were missing. This suggests that the proposed method can be implemented as an alternative to GNSS-based tracking or as a part of a hybrid navigation solution, which reduces the cost and accuracy requirements of the GNSS system.

The remainder of the paper is organized as follows. Section 2 presents some related work. Section 3 states the problem and describes the different elements of the proposed method for structure tracking along vine rows. Section 4 presents the experimental results and the validation against a reliable ground truth in production sites. Finally, Section 5 concludes the paper and provides some ideas for future work.

2 Related work

This paper proposes a navigation approach based on the perception and identification of a geometric structure in the scene. Crop rows provide this structure in the majority of cases where an agricultural robot has to intervene. In this setting, the problem discussed here is related to the field of crop/weed detection and discrimination in the computer vision literature. Crop rows detection is generally a first step where its outputs are used for further analysis in order to provide solutions for weed detection [3], modeling plant characteristics [20], or autonomous navigation [5]. Hence, the discrimination accuracy and distinctiveness requirements of a crop rows detection algorithm varies depending on the application. Mature results have been presented in the literature with the use of deep convolutional neural networks [4] with the objective of increasing detection accuracy and reducing the fault rate for weeding applications and offline planning. On the other hand, for autonomous navigation applications, crop rows are identified as a local feature in a framework that takes the inaccuracy of feature extraction as an input to a model matching and filtering step [5, 17]. In such a framework, a trade-off between accuracy and efficiency takes place. Therefore, high discrimination accuracy and distinctiveness of the algorithm, which comes at the expense of increased computational cost, is not necessarily crucial for this task, and a computationally efficient *accurate-enough* feature extraction technique is often preferred.

Classical vision-based navigation approaches in agricultural environments implement line fitting techniques in order to identify the underlying parallel line structure of crop rows from image data. Common line fitting techniques include linear regression [24], RANSAC [21] and Hough transform [22]. Hough transform is a mathematical transformation between a feature space (often the image space) and a parameter space called Hough space. It is used as a pattern recognition technique to identify a parametric model such as a line through a voting procedure

in the Hough space. Hough transform is preferred over linear regression due to its robustness against outliers and its ability to be generalized to identify curves and complex patterns. A common approach is to apply line fitting on a binary image obtained after a segmentation, done by thresholding appearance features such as color indices [12], aiming to separate pixels belonging to the crops from other elements in the scene: Hough transform was applied in [22] after the image was binarized by thresholding grey values and in [10] after thresholding excess green index (ExG). The simple application of the Hough transform to detect a single line reaches a limitation in situations with significant presence of weed or high vegetation density. To deal with such situations, search techniques in the Hough space that exploit prior hypotheses about the field were introduced. Row spacing was used in [2] to search for parallel lines in the Hough space. Row parallelism was exploited in [14, 10], and the authors concluded that, in Hough space, the peaks corresponding to parallel lines are aligned around the overall orientation.

A different pattern analysis algorithm was proposed in [8]. The authors proposed a segmentation-free image-based method to extract the heading of parallel crop rows by analyzing the texture in a grayscale image. In this method, the image is first wrapped into an overhead view. Then, the image is iteratively skewed, and the variance of the sum of grey values for each column is computed at each iteration. Finally, the overall heading of the structure is considered to be the angle that maximizes the computed variance. This method was completed in [9] with a learning method to learn a crop model that was used to calculate the lateral deviation.

The preprocessing steps in a typical image-based pipeline perform color analysis in order to find a set of keypoints on which line fitting will be applied. Other operations are required in order to project the keypoints in a 3D metric space relative to the guided vehicle. These operations introduce challenges such as dealing with variable appearance due to seasonal changes and light conditions. LiDAR sensors, on the other hand, provide precise measurements directly in that space, and through a simple preprocessing (e.g., height thresholding), the set of keypoints to be used to detect crop rows can be found. This was applied in [11], where the Hough transform was applied on data points acquired by two laser scanners in order to guide a vehicle between tree rows in an orchard. In [21], the RANSAC algorithm was applied on 2D laser-scan measurements to detect crop rows at different growth stages. A common keypoints extraction method was discussed in [26] for LiDAR and vision measurements by thresholding the height for point clouds and a color index for images. Furthermore, the Hough transform was extended in [26] to fit a pattern of parallel equidistant lines parametrized by a common orientation, an offset, and spacing. Such parameterization improved the detection, and it requires less prior knowledge about the structure since it includes the spacing in the parameter space. The drawback of this method is that it increases the computation complexity by adding a third dimension to the parameter space.

In this work, LiDAR point clouds are used as the perception modality in order to identify and track a parallel structure. The proposed algorithm applies Hough transform with a parameterization motivated by a practical interpretation of point cloud statistics. Furthermore, a search strategy in Hough space is applied in order to estimate the overall structure heading inspired by the ideas from [10, 8] without adding the third dimension to the search space. Eventually, to provide complete and reliable navigation in challenging environments with high vegetation density,

a model matching and filtering algorithm is developed to deal with the inaccuracy and faults during the feature extraction step.

3 Materials and methods

The method described in this paper is designed and tested on Trektor¹, a 3 tons versatile agricultural robot developed by SITIA. The robot, shown in two vineyard configurations in Figure 1, is designed to intervene on different crops with their variety of features, ranging from tiny plants in market gardening to dense small trees in viticulture. The robot can also adjust its width to adapt to different vineyard configurations: wide and narrow vineyards. A parallel structure is common in all of these fields, whether it is provided by crop rows or man-made structures (e.g., wheel tracks). Thus, it is worthy to investigate a structure tracking solution based on local perception. This section explains the motivation and development of the LiDAR-based structure tracking method.



(a) Narrow vineyard.



(b) Wide vineyard.

Fig. 1: SITIA's Trektor: the robotic platform used in the experimentation.

3.1 Problem Statement and overview of the proposed solution

One of the main applications of our robotic platform is autonomous navigation in vineyards. The current solution implemented onboard requires a global GNSS path covering the entire field. Hence, the robot is equipped with an accurate global navigation system based on the fusion of RTK-GNSS and high accuracy inertial navigation systems. Leveraging this accuracy in practice requires the global path to be in high resolution, meaning that several waypoints along the vine rows need to be collected using a GNSS rover, which requires a significant effort in the task preparation phase. Hence comes the motivation for a local perception navigation method that exploits the navigation clues provided by the crop row structure. In this setting, the accuracy and resolution requirements of the global path are significantly reduced: only the start and endpoints of the desired rows need to be

¹ www.sitia.fr/en/innovation-2/trektor/

surveyed in order to create the global path that handles the transition between rows while leaving accurate row tracking for the local perception algorithm.

Local perception for this task is provided by a 16-channel 3D LiDAR sensor (Ouster OS1-16) mounted on top of the robot, as shown in Figure 2. The design decision behind such positioning is to allow several rows to be perceived in different vineyard configurations as well as in various types of cultures. In this setting, vine rows will be perceived from the top, resulting in the structure profile illustrated in Figure 3. The goal of the proposed approach is to identify this structure from 3D LiDAR point clouds as an overall heading $\Delta\gamma$ and the lateral positions y_i of each perceived row. Hence, crop rows are assumed to be locally parallel in the considered range of measurement of the LiDAR sensor (it was set to 10 meters in experimentation). Consequently, the method builds a model for each row that will serve as a reference path to follow. The final outputs of the method are the relative localization information: the lateral deviation Δl and the angular deviation $\Delta\gamma$. This information is suitable for tracking control laws that are designed for car-like mobile robots. We refer the reader to the approach proposed in [15] for a complete explanation of the control law implemented on our robotic platform. The tracking control law is beyond the scope of this paper since the main focus here is on the perception part of the problem.



Fig. 2: LiDAR sensor's positioning.

A schematic overview of the proposed solution is provided in Figure 4 with three major blocks:

- *Preprocessing*: the operations in this step are tailored by the target culture and sensor choice. The aim here is to extract a set of representative keypoints to be used by the pattern analysis algorithm. In the case of vineyards perceived with a 3D Lidar, downsampling and height thresholding is sufficient to eliminate the ground and extract representative keypoints.

- *Structure identification*: this is the step where frame-to-frame rows detection occurs. The proposed algorithm, based on Hough transform and point cloud histograms, extracts row positions and a common heading from the set of keypoints.
- *Model matching*: this step takes the outputs from the identification step and performs data aggregation and filtering in order to construct a persistent model of each row of the structure.

The remainder of this section explains in detail the different steps of the proposed solution.

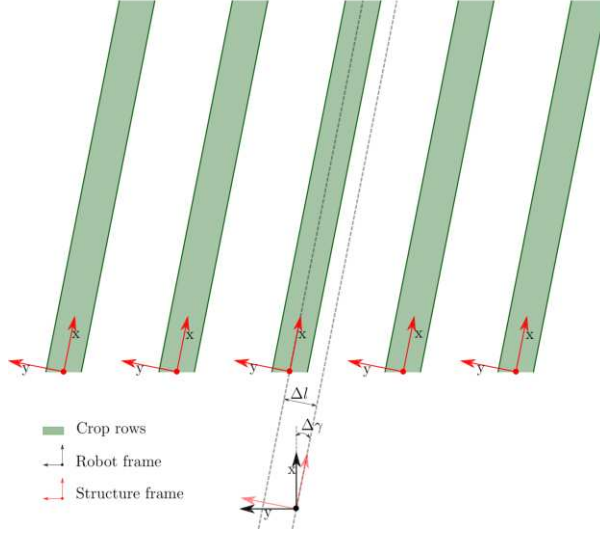


Fig. 3: Illustration of the structure profile to be identified. Tracking is provided by calculating the lateral Δl and angular $\Delta \gamma$ deviations with respect to the reference row.



Fig. 4: Overview of the proposed solution.

3.2 Structure identification

The goal here is to identify a set of parallel rows from input point clouds after preprocessing through height thresholding and voxel grid downsampling [23].

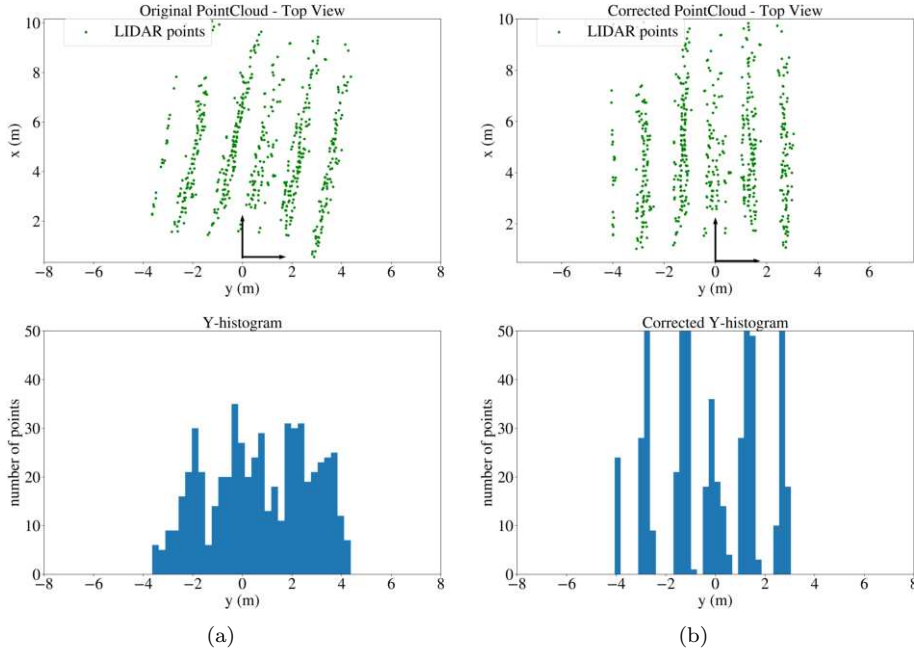


Fig. 5: The top view of two point clouds along with the corresponding y-histograms. (a) A point cloud with initial heading with respect to the robot's frame and its y-histogram. (b) The corrected point cloud after iterative skewing (zero heading) and its y-histogram. The y-histogram of the corrected point cloud has a maximal variance.

3.2.1 Heading estimation with iterative skewing

Figure 5a shows an example of an input point cloud acquired in a narrow vineyard (configuration shown in Figure 17a). The core of the proposed structure identification method is the estimation of a common heading by analyzing the entire input point cloud: Figure 5 shows the concept behind heading estimation. The first row shows illustrations in the x-y plane of the same point cloud except that in Figure 5b, the point cloud is rotated in a way to be aligned with the x-axis. The second row shows the corresponding y-histograms of the point clouds. It is clear that the histogram of the corrected point cloud has peaks that can identify each row, while the histogram of the original point cloud is more flattened, and the rows cannot be separated. Hence, the information about the heading can be resumed in the y-histogram: the variance of histogram values is maximized when the point cloud has zero heading [8]. The method used here to maximize this variance, and thus, finding the common heading, is the one originally proposed in [8] for images where the equivalent of y-histogram was the sum along the columns of a grayscale image. This method, named iterative skewing, consists of iteratively rotating the cloud about the z-axis while computing the y-histogram along with its variance in each iteration. Finally, the heading is considered to be the angle that maximizes the variance.

3.2.2 Relation to the Hough transform

Looking into the iterative skewing algorithm, the relationship between the coordinates of a point in the rotated cloud and the original cloud is the following:

$$y_{rot} = x_0 \sin \gamma + y_0 \cos \gamma \quad (1)$$

$$= x_0 \cos \theta + y_0 \sin \theta, \quad \gamma = \frac{\pi}{2} - \theta, \quad (2)$$

where γ is the heading rotation angle, y_{rot} is the ordinate of a point in the rotated cloud, and (x_0, y_0) are the coordinates of the corresponding point in the original cloud. Equation 2, when generalized to any (x, y) tuple, is the representation of a line according to the *normal parameterization* using the normal angle θ and a signed distance y_{rot} which is the standard representation of a candidate line in Hough transform [7]. Therefore, computing the histogram of y_{rot} for iterative values of γ is equivalent to computing the Hough transform of the point cloud through a 2D voting procedure.

Indeed, Hough transform maps input points from their 2D metric space (x, y) to the parameter space (i.e., the Hough space) (y_{rot}, γ) . A point (x_0, y_0) in the input point cloud is mapped to a sinusoid in the Hough space (y_{rot}, γ) that is defined in Equation 2. This sinusoid defines the parameter of all the possible lines that pass through the point (x_0, y_0) . Thus, each point *votes* for the parameter of candidate lines. After considering all points, the Hough representation (also called the Hough accumulator) of the input cloud is a heatmap in the 2D parameter space where the coordinates of high heat define the most likely model candidates. In this setting, each column of the resulting Hough accumulator is the y-histogram of the point cloud rotated by a heading γ .

Such an interpretation gives a formal understanding of the iterative skewing algorithm as a search method in Hough space that estimates the heading angle by finding the column with maximal variance. This search method finds the common heading of a pattern of parallel lines without integrating it in the search space. Therefore, the search space remains limited to only two parameters: (y_{rot}, γ) . This provides a considerable advantage in terms of computational time by not adding a third dimension to the search space as in other pattern fitting methods [26]. Figure 6 illustrates the heading estimation method: the Hough accumulator for $\gamma \in [-20^\circ, 20^\circ]$. The plot of column variance with respect to the heading angle is shown in the second row along with the estimated pattern heading.

3.2.3 Detecting lateral row positions

Once the heading is estimated, each row in the structure must be located (position on the y-axis) in order to complete the crop profile. This is done through 1D peak detection on the accumulator column corresponding to the estimated heading. This column is the y-histogram of the corrected point cloud in Figure 5b and thus finding its peaks would locate each row on the y-axis. To this end, a simple peak detection method is proposed to complete the detection. The proposed peak detection method is efficient and does not require prior knowledge about the structure details such as spacing and row width (these are often required in standard peak detection methods such as mean shift [6] or k-means). Figure 7 shows the principle of the algorithm. First, a thresholding step is performed on

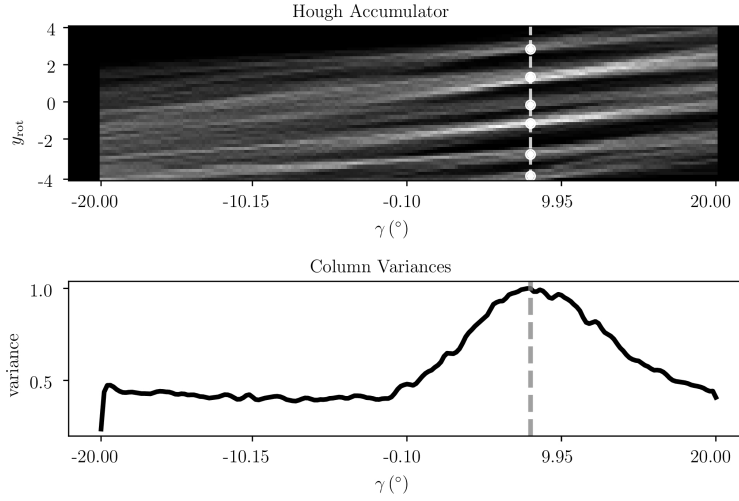


Fig. 6: Illustration of the heading estimation method in Hough space. First row: the hough accumulator of the input point cloud. Second row: the column variance with respect to the heading angle. The heading is estimated as the angle that maximizes the variance.

histogram values (using their mean as the threshold) to find histogram segments that represent interval estimation of the position of each row. Then, a position is estimated from each interval as the mean of its values. Since the input point cloud corresponds to the crop's canopy, the exact position of the tree trunk is not directly observable and thus needs to be estimated. The mean of histogram values provides a reasonable estimate of the trunk's position as the center of the perceived canopy row. The results of peak detection are presented in Figure 6 as white dots in the Hough accumulator.

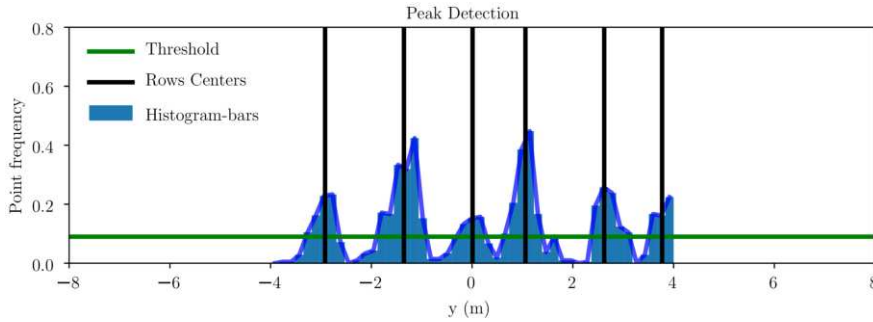


Fig. 7: Illustration of the peak detection algorithm used to find the y-coordinate of each row.

In summary, the structure identification step applies Hough transform with a custom search method in the parameter space in order to identify a common heading along with the position of each row. Imperfections in the detection could occur, such as discontinuity and inaccuracy of row position estimation due to high vegetation pressure in the discussed case. The discretization resolution of the parameters in Hough space also produces some inaccuracy. This is not ideal for accurate navigation since the tracking algorithm requires continuous and smooth lateral and angular deviations. Therefore, a model matching and filtering step is required in order to construct a consistent and robust model to be used as a reference path.

3.3 Model matching

At each time step and as the robot moves forward, structure identification estimates, for each point cloud, an overall heading γ and a set of row positions. Thus providing local observations as a set of points

$$O_t = \{(x_i, y_i, \gamma_i), \quad i : 1 \rightarrow n\} \quad (3)$$

where x_i and y_i represent the coordinates of a detected row center, γ_i is the corresponding estimated heading, and n is the number of rows detected per input frame (point cloud). These observations are not consistent in their representation of the actual structure elements (crop rows): discontinuity occurs in these observations due to gaps in the vegetation, which is common in old production sites, or due to detection failures. Figure 8 illustrates the general idea behind the model matching step in such situations. The goal here is to *make sense* of the local observations by associating them with a unique model of each row that is persistent across all the frames. The models constructed in this step are linear regression models representing line equations in a relative frame. Thus, an odometry localization is needed to provide this relative frame.

A detailed diagram of the operations in the model matching step is provided in Figure 9. The first set of observations is used to initialize line models. After that, the existing models are used to associate the frame-to-frame centers. Data association here means attributing each center (x_i, y_i, γ_i) to a unique model ($l^{(id)}$). Finally, the associated centers are used to update the existing models and create new ones if needed. These three operations are explained in detail below.

Initialization: The initialization step is crucial since data association and model updating are highly dependent. A poor initialization could lead to failure in the following operations. Here it is simply supposed that the algorithm will be activated in front of the structure² so that the first observation O_1 would be reliable. Once the first set of observations O_1 is perceived, row models are initialized accordingly. A row model ($l^{(id)}$), defined with the parameters $w_0^{(id)}$ and $w_1^{(id)}$, is initialization per each (x_i, y_i, γ_i) data point in O_1 as follows:

$$(l^{(id)}) : y = x \tan \gamma_i + y_i - x_i \tan \gamma_i := w_1^{(id)} x + w_0^{(id)}. \quad (4)$$

² Since the presence of a structure in the perceived point cloud is a base assumption for the proposed algorithm; a global planner is needed to ensure this condition at row starts, as will be discussed in section 4.

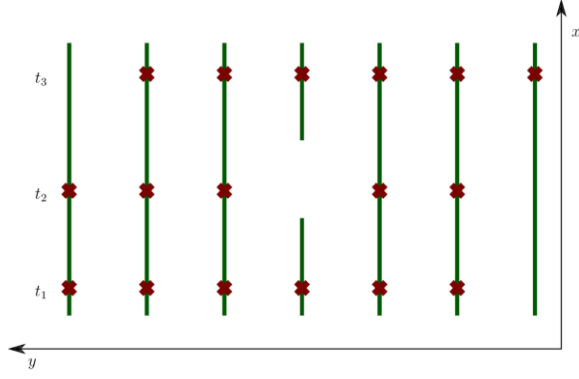


Fig. 8: Illustration of the goal of model matching. Three cases are presented to show that local observations are not necessarily consistent with their representation of the actual rows: a missing detection, a gap, or a row that appears after several time steps. The detected row centers (red dots) are to be used to build a regression model for each row in a relative frame.

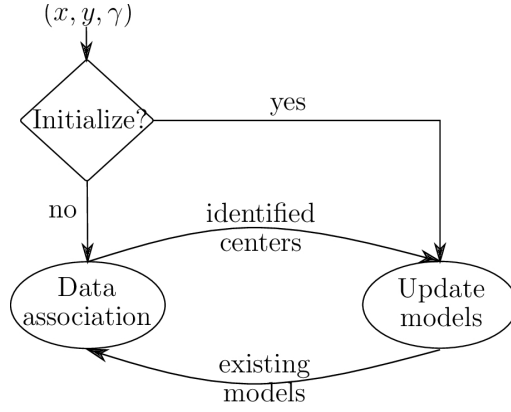


Fig. 9: A detailed diagram of the model matching step. Frame-to-frame row centers are used to initialize and build regression models. This is done through a cycle of associating centers with existing models and updating the models using the associated centers.

In the case of Figure 3, five models would be initialized according to Equation (4) as five lines placed at rows positions (x_i, y_i) and oriented with the common heading γ_i .

Data association: After row models initialization, tracking begins by calculating lateral and angular deviations with respect to the selected row model to follow. As the robot moves forward, row centers from frame-to-frame profiles are mapped in the relative frame and should be assigned to row models in order to update them. The assignment is done using a simple zone rule based on the distance d_{ij} , between frame-to-frame row centers (x_i, y_i) and existing models $(l^{(j)})$, and a row spacing

s that is inferred as the median of the differences between successive intercepts of existing models. A greedy approach is considered; therefore, the best candidate model for the input (x_i, y_i) is the model $(l^{(j_{min})})$ with the lower distance. The following zone rule is used to decide whether center (x_i, y_i) has to be assigned to $(l^{(j_{min})})$, rejected, or used to initialize a new model:

$$\begin{cases} d_{ij} < 0.4s & \Rightarrow \text{assign} \\ 0.4s < d_{ij} < 0.5s & \Rightarrow \text{reject outlier} \\ d_{ij} > 0.5s & \Rightarrow \text{initialize new model} \end{cases}$$

Updating models with regularized regression: Data points in the data association step are assigned to the buffer of the corresponding row model as a set of (x_i, y_i, γ_i) points. A data buffer is used to update a row model using linear regression. Linear regression is formalized as follows:

$$(l^{(id)}) : \mathbf{Y} = \mathbf{X}\mathbf{W}^{(id)} \quad (5)$$

where

$$\mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \text{ and } \mathbf{W}^{(id)} = \begin{bmatrix} w_0^{(id)} \\ w_1^{(id)} \end{bmatrix}.$$

As mentioned in the explanation of the structure identification step, rows centers (x_i, y_i) estimation is not as accurate as heading γ_i estimation. Thus, in the early steps of tracking, when there are only a few data points along row directions, *classic* linear regression would lead to false lines and therefore, data association would fail, leading to a failure in tracking.

In order to initialize robust regression models, the heading—which represents strong information about the row direction—is added as a regularization term for the linear regression. The heading is integrated as the regularization term of the following loss function³:

$$L = \sum_{i=1}^n \left(y_i - \mathbf{x}_i^T \mathbf{W} \right)^2 + \alpha \sum_{i=1}^n |\vec{n} \cdot \vec{\gamma}_i|^2 \quad (6)$$

$$= \sum_{i=1}^n \left(y_i - \mathbf{x}_i^T \mathbf{W} \right)^2 + \alpha \sum_{i=1}^n (\tan \gamma_i - w_1)^2 \quad (7)$$

$$= \|\mathbf{Y} - \mathbf{X}\mathbf{W}\|^2 + \alpha \|\Gamma - \mathbf{a}\mathbf{W}\|^2, \quad (8)$$

where $\Gamma = [\tan \gamma_1 \ \dots \ \tan \gamma_n]^T$, $\mathbf{a} = \begin{bmatrix} 0 & \dots & 0 \\ 1 & \dots & 1 \end{bmatrix}^T$ and α is the regularization parameter. Minimizing the loss function L leads to:

$$\hat{\mathbf{W}} = (\mathbf{X}^T \mathbf{X} + \alpha \mathbf{a}^T \mathbf{a})^{-1} (\mathbf{X}^T \mathbf{Y} + \alpha \mathbf{a}^T \Gamma). \quad (9)$$

The regularization term $\sum_{i=1}^n |\vec{n} \cdot \vec{\gamma}_i|^2$ is the dot product of the normal vector of the solution $\vec{n} = (-w_1 \ 1)$ and the vector $\vec{\gamma} = (1 \ \tan \gamma)$ oriented along the

³ The superscript (id) is omitted in the following since the same optimization is done per row model.

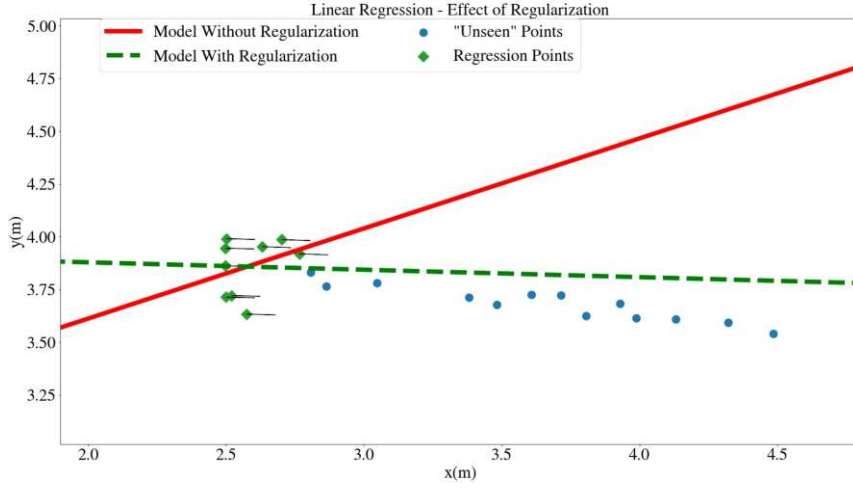


Fig. 10: Effect of using the heading as a regularization term in linear regression using (x_i, y_i, γ_i) data points. Only diamond-shaped points (x_i, y_i) are used in the regression, the small arrows represent the heading γ_i ; blue points are points belonging to the same row and are unseen at the time of regression.

heading angle γ . This regularization penalizes the solutions that are not coherent with the estimated heading. Figure 10 shows the effect of the regularization term on the results. In the presented case and the subsequent experimentations, an aggressive regularization was chosen ($\alpha = 1.5$). This regularization has the most impact after initialization since it ensures that the row model is oriented along the estimated heading γ_i and mitigates the effect of lateral imprecisions due to the estimation of row positions (x_i, y_i) from a discretized histogram. However, this regularization will have less impact in the later stages of tracking with the accumulation of (x_i, y_i) data points along the row direction, which provides more reliable information about the row's direction.

To summarize the overall solution: Point clouds are perceived and preprocessed at the input of the identification step that provides frame-to-frame local observations of the structure. These observations are mapped in the relative frame, and the model matching step build row models from the local observation. Tracking is done by sending lateral and angular deviations with respect to the reference model. Figure 11 shows the results of the combined components of the proposed algorithm. Data association is shown as the color assigned to each center (a unique color per row), and the linear regression models are plotted as the lines representing each row.

4 Experimental evaluation

The proposed algorithm was tested on a vineyard production site in the two configurations presented in Figure 1: wide and narrow vineyards. The baseline solution

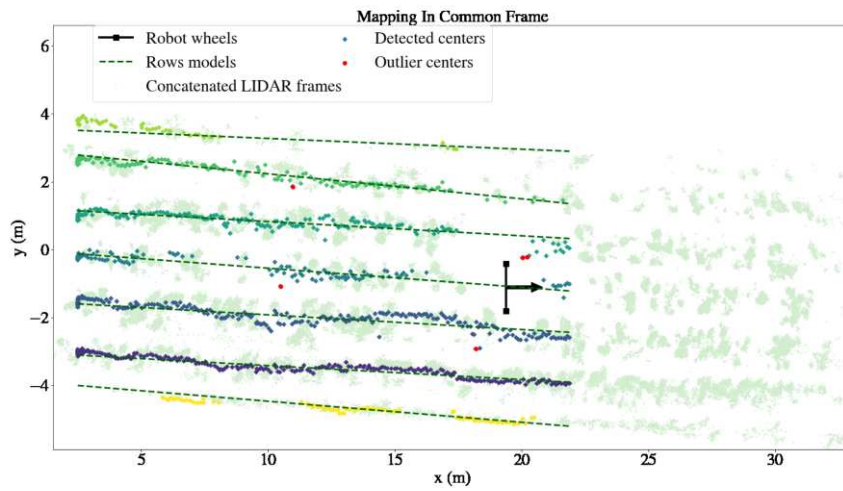


Fig. 11: Mapping of LiDAR frames along with the detected row centers and the corresponding models in a relative frame. The robot is shown in a position where there is a missing tree in the row, showing that tracking continues with respect to the model.

for the navigation task is provided via a reliable GNSS-based localization and navigation system. This solution consists of creating a global path by manually defining geo-referenced points along the rows using a reliable GNSS rover (Septentrio Altus NR3). Figure 12 shows the satellite image of the narrow vineyard (provided by google earth) along with the geo-referenced points and the reference path. In the following, the ground truth for the structure algorithm is considered to be the global GNSS path.



Fig. 12: Satellite image of the vineyard test field along with the geo-referenced points used to create the ground-truth path.

The goal here is to validate the feasibility of the proposed local navigation approach in this perceptually challenging environment. Perception challenges in vineyards come from the irregular shape and density of the perceived canopy. Hence, the experiments were conducted with the goal of validating the consistency and reliability of the proposed algorithm's estimates against those of the baseline GNSS-based solution. To this end, the autonomous tracking was provided by the ground truth GNSS-based algorithm, and the structure-based algorithm was run as an observation in real-time during the experiments. In order to deal with the overhead turns and the transition from one row to another, a global planner was used to define two sub-portions of the path: an in-row portion and an overhead turn portion. The structure-based observations were only activated in the in-row portion of the path, which was shifted 3 meters before the start and end of each row in order to adapt to the measurement range of the LiDAR sensor.

The configuration and parameters were considered in experimentation: point clouds were acquired using a 16-channel LiDAR sensor (Ouster OS1-16) at a frequency of 10 Hz. Preprocessing consisted of cropping the point clouds in the range of $[-4m, 4m]$ along the y-axis in the robot's frame (Figure 3) followed by a voxel grid downsampling with a voxel size of $0.15m$. Y-histograms for heading estimation using iterative skewing were computed by discretizing the $[-4m, 4m]$ interval into 100 bins. For model matching and linear regression, a buffer size of 50 was used with a regularization parameter $\alpha = 1.5$.

4.1 Validation on the wide vineyard

The first experiment consisted of navigating in the wide vineyard (Figure 17b). The navigation was done on eight rows characterized by an inter-row spacing of 1.9m and a row length of approximately 120m. The robot speed was 1 m/s in the in-row zone.

Figure 13 offers a global view of the results of the navigation on eight rows. It shows the ground truth predefined GNSS global path along with the path estimated by the structure-based algorithm. Row limits are also shown on the figure to indicate where the structure algorithm was activated/deactivated. Such illustration on the global scale shows that the path estimated from local structure perception remained consistently available and reliable on the whole covered area.

A more detailed view of the results on one row can be found in Figure 14, with the lateral and angular deviations estimated by the structure algorithm and the ground truth GNSS algorithm. By considering the GNSS-based algorithm as the ground truth of the experiment, the absolute difference between the deviations of the two algorithms is, therefore, the total error of the structure algorithm with respect to the ground truth. In the case presented in Figure 14, the redundancy of the two algorithms can be concluded with a slight lateral offset. The root mean square error (RMSE) characterizing the lateral offset of the algorithm on all the rows is summarized in Table 1. The average RMSE for this experiment is 0.107m which is reasonable compared to 1.9m row spacing. This lateral offset is investigated in the next experiment presented in the following section.

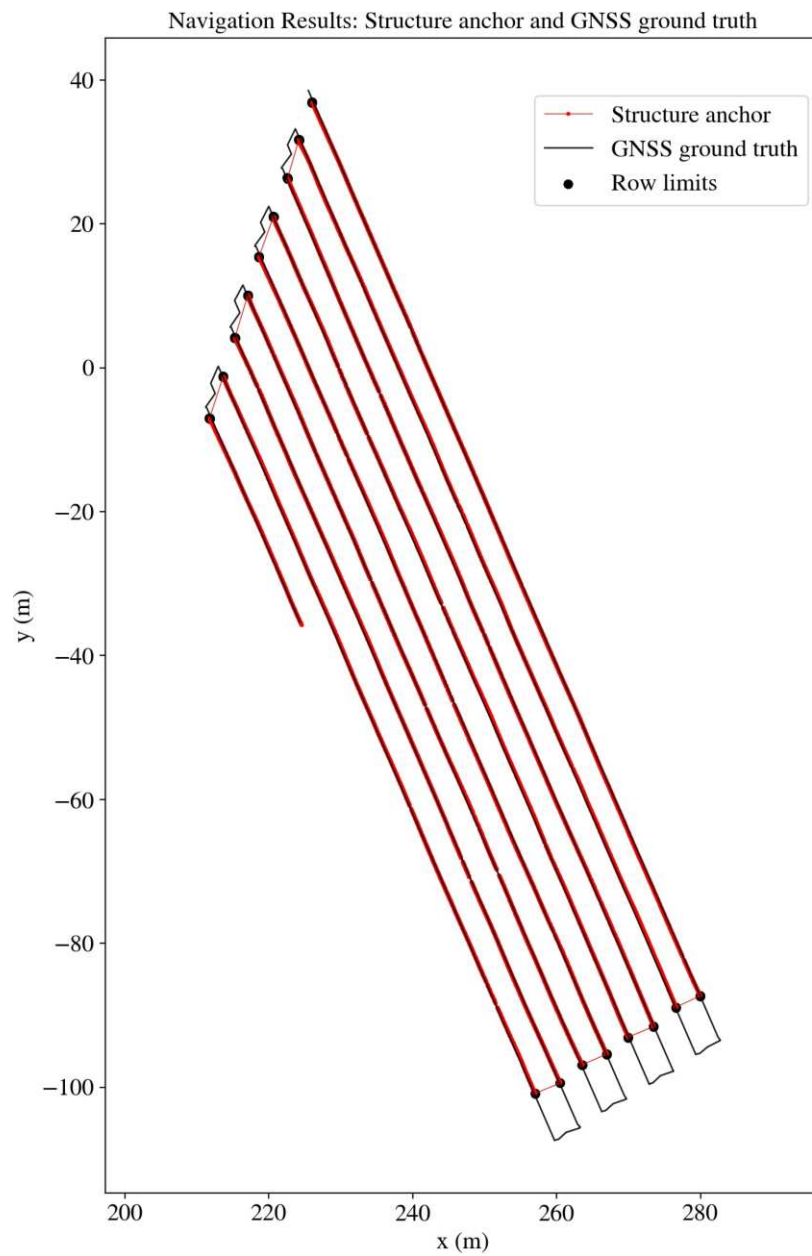


Fig. 13: The ground truth GNSS path and the estimated structure path. The rows do not have the same length. The black dots indicate the positions on which the structure tracking algorithm was activated/deactivated.

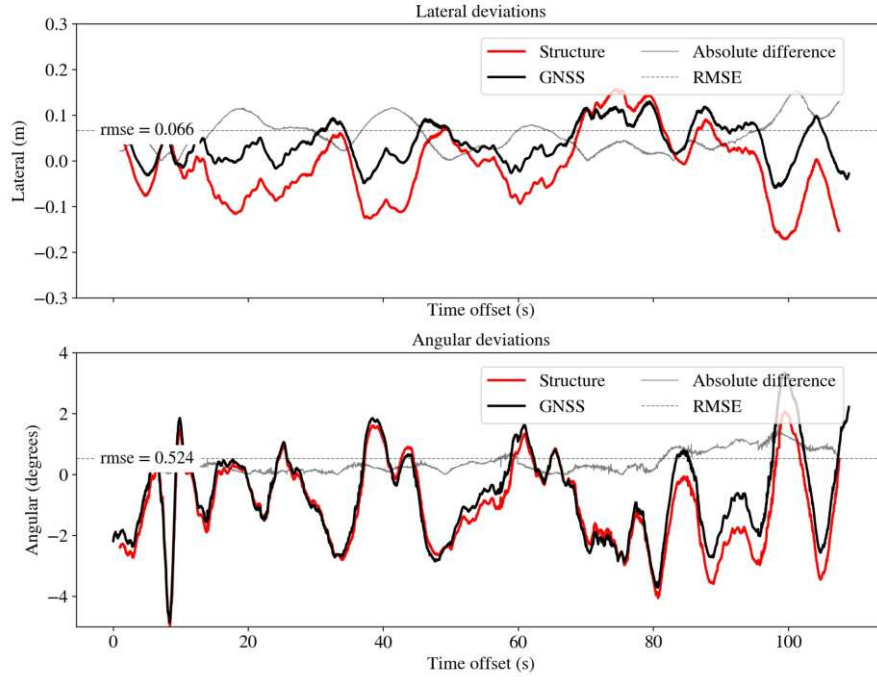


Fig. 14: The lateral and angular deviations estimated by both the proposed structure-based and the baseline GNSS-based algorithms. The absolute difference between the deviations along with the root mean square error is noted on each graph.

Row number	r1	r2	r3	r4	r5	r6	r7	r8	average
RMSE lateral (cm)	6.6	14.4	4.6	13.4	7.9	16.7	10.7	10.9	10.7

Table 1: Root mean square lateral errors on each row of the wide vineyard.

4.2 Validation on the narrow vineyard

The same experiment was repeated on seven rows of the narrow vineyard (Figure 17a). The RMSE lateral error for this experiment is reported in Table 2 with slightly higher values than the first experiment.

Row number	r1	r2	r3	r4	r5	r6	r7	average
RMSE lateral (cm)	6.4	13.3	13.1	12.6	7.5	20	11.2	12

Table 2: Root mean square lateral errors on each row of the narrow vineyard.

To get an insight into the repeatability of the results and to understand the higher RMSE on some rows, a second experiment was conducted on one row of the narrow vineyard and consisted of repeating the navigation five times on the same

row. Figure 15 shows the concatenation of the point clouds from each frame along with the GNSS reference and the estimated structure path for all five passages. It can be concluded from the figure that the estimated structure path follows the perceived vegetation, which is the vine canopy. The position and shape of the canopy with respect to the trunks varies depending on the tree. In some situations, it is wider and can be shifted on one side of the trunk, as it is shown in the middle of the concerned row in Figure 15.

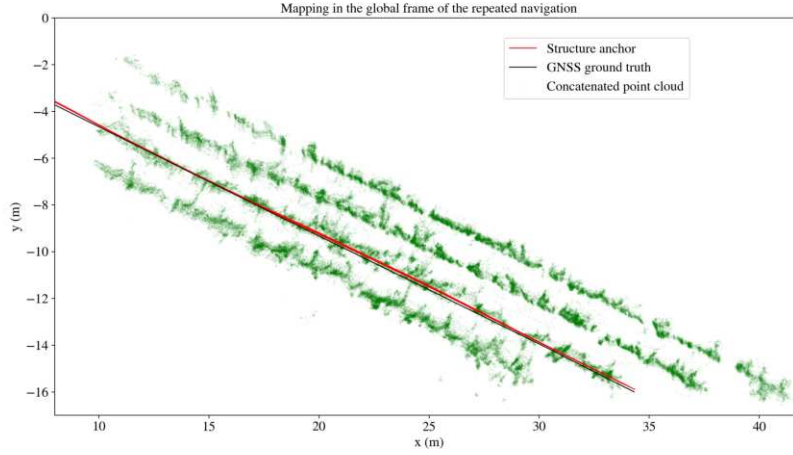


Fig. 15: The estimated structure path of the repeated navigation along with the ground truth GNSS path and the concatenated point clouds.

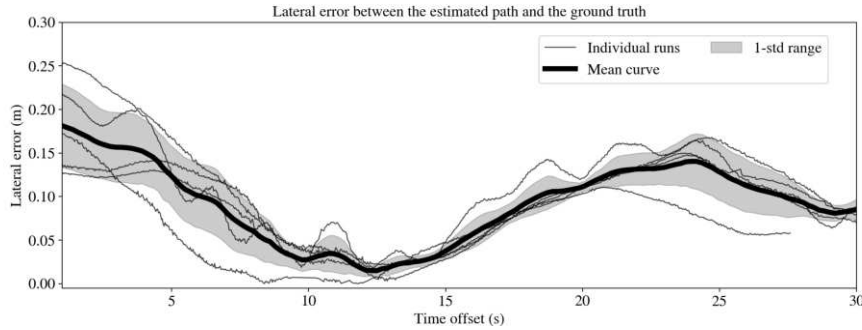


Fig. 16: The lateral error between the estimated structure path and the ground truth of each of the individual runs along with their mean curve and the 1-std range.

This can be further noticed in Figure 16 that shows the lateral error between the estimated structure path and the ground truth GNSS path of all five passages. The lateral error is plotted with respect to the time offset since each observation

was activated on the same location at the beginning and with the same speed profile. The path estimation of each run starts with a lateral error of approximately 0.20m that decreases afterward to under 0.05m. This offset at the beginning is justified since only a few noisy frame-to-frame observations were received, and the regression model is less reliable with fewer data at the start. This can also be seen in the standard deviation of the offset across all the runs that decreases as the regression model gathers more data (the height of the 1-std range shown on the figure); a lower standard deviation indicates the repeatability of the algorithm. The lateral offset increases again towards the end; however, by comparing this offset with the results reported in Figure 15, this can be interpreted by the shift and width of the perceived canopy with respect to the ground truth. Figure 17 shows two photos of the canopy taken by an onboard camera on the robot during the experiment: one of a *regular* canopy and another of a wide canopy that scored an offset of 0.20m with respect to the ground truth.



Fig. 17: Photos taken by an onboard camera on the robot in the narrow vineyard. (a) a *regular* vine canopy, (b) a wider vine canopy that caused the estimated structure path to shift 0.20m from the ground truth.

Finally, the performed experiments demonstrate a redundant behavior of the proposed structure-based algorithm with respect to a reliable GNSS-based algorithm. The peak lateral offset in the most challenging cases remained under 0.2m; this ensures satisfying autonomous navigation in real scenarios knowing that the inter-row separation of the vineyard structure is between 1.5 and 1.9 meters. The main practical advantage of the proposed algorithm is that it does not require a prior manually defined global path and provides robust and similar tracking as the GNSS-based tracking. A practical approach that reduces the effort in the task preparation phase is to integrate the proposed algorithm in a hybrid approach with a GNSS-based solution that defines only the row limits and the overhead turns and uses the structure-based algorithm to navigate along the rows of the crop.

5 Conclusion and perspectives

This paper presents a robust LiDAR-based method for autonomous structure tracking in vineyards; which is one of the most challenging structure tracking scenarios in precision agriculture due to the wide and dense vine canopy. The proposed method does not require prior field surveying, and it is insensitive to crop-specific characteristics such as row width and spacing. In fact, the proposed method is based on the identification of the navigation clues provided by crop row structures. This allowed the proposed algorithm to be suitable for a versatile robot that intervenes in different types of cultures. Indeed, it was shown that the proposed method provides accurate and robust tracking in two vineyard configurations: wide and narrow vineyards. This was achieved by exploiting a practical interpretation of point cloud histograms to extract local structure observations from Hough space that are used to build an online crop model as the robot moves forward in the field. A key challenge while building an online model is proper initialization. Hence, robust heading estimation used as a regularization term provides a strong initialization for linear regression models.

Experimental validation was conducted on vineyards. The experiments confirm a robust structure tracking that is redundant with a reliable GNSS-based solution while significantly reducing the requirements of the fastidious task preparation phase (only the row limits were used to handle the transition from one row to another). Moreover, the proposed method can reduce the accuracy requirements, and thus the cost, of the GNSS subsystem since it provides accurate tracking relative localization with respect to the perceived structure.

A future work direction is to study the behavior of the proposed algorithm on different types of cultures, such as greenhouses and open fields. The idea is to consider wheel tracks as *negative rows* after extracting the ground plane from the point cloud using a random sample consensus algorithm. Hence, further analysis is required to study the consistency of the proposed algorithm in such fields and to define suitable conditions for applicability such as minimum plant height or wheel track depth. By doing so, the proposed approach could contribute to providing full autonomy on various production sites, including areas with poor precision GNSS coverage.

Acknowledgements This work is partially supported by the *Agence Nationale de la Recherche* ANR through the ROSE challenge, ANR-17-ROSE-0002-01 and by the *Association Nationale de la Recherche et de la Technologie* ANRT through the CIFRE PhD, ANRT CIFRE N° 2018/0792.

Declarations

Funding This work is partially supported by the *Agence Nationale de la Recherche* ANR through the ROSE challenge, ANR-17-ROSE-0002-01 and by the *Association Nationale de la Recherche et de la Technologie* ANRT through the CIFRE PhD, ANRT CIFRE N° 2018/0792.

Conflicts of interest : The authors have no conflicts of interest to declare that are relevant to the content of this article.

Availability of data and material : None.

Code availability : None.

Authors' contributions : All authors contributed to the study's methodology, conception and design. Formal analysis: Hassan Nehme; Software and experimental validation: Hassan Nehme, Clément Aubry, Thomas Solatges; Resources: Hassan Nehme, Clément Aubry, Thomas Solatges; Writing: The first draft of the manuscript was written by Hassan Nehme and all authors commented on previous versions of the manuscript, all authors read and approved the final manuscript; Supervision and administration: Clément Aubry, Romain Rossi, Rémi Boutteau; Funding acquisition: Thomas Solatges, Xavier Savatier.

Ethics approval : Not applicable.

Consent to participate : Informed consent was obtained from all participants in the study.

Consent for publication : Participant consented to the submission of this article to the journal.

References

1. Aly, M.: Real time detection of lane markers in urban streets. In: 2008 IEEE Intelligent Vehicles Symposium, pp. 7–12. IEEE (2008). DOI 10.1109/IVS.2008.4621152
2. Åstrand, B., Baerveldt, A.J.: A vision based row-following system for agricultural field machinery. *Mechatronics* **15**(2), 251–269 (2005). DOI 10.1016/j.mechatronics.2004.05.005
3. Bah, M.D., Hafiane, A., Canals, R.: Weeds detection in uav imagery using slic and the hough transform. In: 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA), pp. 1–6. IEEE (2017). DOI 10.1109/IPTA.2017.8310102
4. Bah, M.D., Hafiane, A., Canals, R.: Crownnet: Deep network for crop row detection in uav images. *IEEE Access* **8**, 5189–5200 (2019). DOI 10.1109/ACCESS.2019.2960873
5. Basso, M., de Freitas, E.P.: A uav guidance system using crop row detection and line follower algorithms. *Journal of Intelligent & Robotic Systems* **97**(3), 605–621 (2020). DOI 10.1007/s10846-019-01006-0
6. Cheng, Y.: Mean shift, mode seeking, and clustering. *IEEE transactions on pattern analysis and machine intelligence* **17**(8), 790–799 (1995). DOI 10.1109/34.400568
7. Duda, R.O., Hart, P.E.: Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM* **15**(1), 11–15 (1972). DOI 10.1145/361237.361242
8. English, A., Ross, P., Ball, D., Corke, P.: Vision based guidance for robot navigation in agriculture. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 1693–1698. IEEE, Hong Kong, China (2014). DOI 10.1109/ICRA.2014.6907079
9. English, A., Ross, P., Ball, D., Upcroft, B., Corke, P.: Learning crop models for vision-based guidance of agricultural robots. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1158–1163. IEEE, Hamburg, Germany (2015). DOI 10.1109/IROS.2015.7353516
10. Gée, C., Bossu, J., Jones, G., Truchetet, F.: Crop/weed discrimination in perspective agronomic images. *Computers and Electronics in Agriculture* **60**(1), 49–59 (2008). DOI 10.1016/j.compag.2007.06.003
11. Hamner, B., Singh, S., Bergerman, M.: Improving orchard efficiency with autonomous utility vehicles. In: 2010 Pittsburgh, Pennsylvania, June 20–June 23, 2010, p. 1. American Society of Agricultural and Biological Engineers (2010). DOI 10.13031/2013.29902

12. Hamuda, E., Glavin, M., Jones, E.: A survey of image processing techniques for plant extraction and segmentation in the field. *Computers and Electronics in Agriculture* **125**, 184–199 (2016). DOI 10.1016/j.compag.2016.04.024
13. Hiremath, S.A., Van Der Heijden, G.W., Van Evert, F.K., Stein, A., Ter Braak, C.J.: Laser range finder model for autonomous navigation of a robot in a maize field using a particle filter. *Computers and Electronics in Agriculture* **100**, 41–50 (2014)
14. Jones, G., G  e, C., Truchetet, F.: Modelling agronomic images for weed detection and comparison of crop/weed discrimination algorithm performance. *Precision Agriculture* **10**(1), 1–15 (2009). DOI 10.1007/s11119-008-9086-9
15. Lenain, R., Thuilot, B., Cariou, C., Martinet, P.: Adaptive and predictive non linear control for sliding vehicle guidance: Application to trajectory tracking of farm vehicles relying on a single rtk gps. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566), vol. 1, pp. 455–460. IEEE (2004). DOI 10.1109/IROS.2004.1389394
16. Merriaux, P., Dupuis, Y., Boutteau, R., Vasseur, P., Savatier, X.: Robust robot localization in a complex oil and gas industrial environment. *Journal of Field Robotics* **35**(2), 213–230 (2018). DOI 10.1002/rob.21735
17. Nagham, S., Tobias, L., Cheryl, M., Nigel, H.: A review of autonomous navigation systems in agricultural environments. *Innovative Agricultural Technologies for a Sustainable Futu* pp. 22–25 (2013)
18. Ort, T., Paull, L., Rus, D.: Autonomous vehicle navigation in rural environments without detailed prior maps. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 2040–2047. IEEE (2018). DOI 10.1109/ICRA.2018.8460519
19. Pan, X., Shi, J., Luo, P., Wang, X., Tang, X.: Spatial as deep: Spatial cnn for traffic scene understanding. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
20. Rabab, S., Badendorst, P., Chen, Y.P.P., Daetwyler, H.D.: A template-free machine vision-based crop row detection algorithm. *Precision Agriculture* pp. 1–30 (2020). DOI 10.1007/s11119-020-09732-4
21. Reiser, D., Miguel, G., Arellano, M.V., Griepentrog, H.W., Paraforos, D.S.: Crop row detection in maize for developing navigation algorithms under changing plant growth stages. In: Robot 2015: Second Iberian Robotics Conference, pp. 371–382. Springer (2016). DOI 10.1007/978-3-319-27146-0_29
22. Rovira-M  s, F., Zhang, Q., Reid, J.F., Will, J.D.: Hough-transform-based vision algorithm for crop row detection of an automated agricultural vehicle. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* **219**(8), 999–1010 (2005). DOI 10.1243/095440705X34667
23. Rusu, R.B., Cousins, S.: 3D is here: Point Cloud Library (PCL). In: IEEE International Conference on Robotics and Automation (ICRA). Shanghai, China (2011). DOI 10.1109/ICRA.2011.5980567
24. S  gaard, H.T., Olsen, H.J.: Determination of crop rows by image analysis without segmentation. *Computers and electronics in agriculture* **38**(2), 141–158 (2003). DOI 10.1016/S0168-1699(02)00140-0
25. Tipaldi, G.D., Meyer-Delius, D., Burgard, W.: Lifelong localization in changing environments. *The International Journal of Robotics Research* **32**(14), 1662–1678 (2013). DOI 10.1177/0278364913502830
26. Winterhalter, W., Fleckenstein, F.V., Dornhege, C., Burgard, W.: Crop Row Detection on Tiny Plants With the Pattern Hough Transform. *IEEE Robotics and Automation Letters* **3**(4), 3394–3401 (2018). DOI 10.1109/LRA.2018.2852841