Tracking footprints for agricultural applications: a low cost lidar approach

Thibault Tourrette[1], Roland Lenain[1], Raphael Rouveure[1] and Thomas Solatges[2]

***Abstract*— The accuracy of a mobile robot is an important topic, which is subjected to an important attention. It is particularly crucial when considering off-road applications such as agricultural robot, since it rises questions for both localization and control. A possible way to avoid the use of expensive sensors (such as RTK-GPS) or building specific landmarks, lies in the exploitation of tire footprints, let in the soil by previous actions achieved manually or autonomously. These footprints are particularly detectable by several kind of sensors. This paper presents a method for tracking footprints with a low cost lidar sensor. The proposed way is to detect an expected footprint template and then aggregate successive detections to construct a 3D map. This map finally permit to compute a control law for path tracking regarding mobile robot position compared with footprint location.**

I. INTRODUCTION

Agricultural robotics is a promising field of application, as it permits reducing arduousness, and risks for humans, while ensuring a high level of production [3]. Another interest of agricultural robotics is to reduce chemical products usage. Currently, mechanical solutions are often too expensive because they need more time than chemical solutions, in that way an autonomous robot could appear as a good idea. As a result, research is paying more and more attention to this area, leading to important advances in terms of perception and control [1]. In particular, the problem of path tracking is deeply studied. It is particularly important considering farm applications, since robots have to move following the vegetation rows. Many solutions are based RTK-GPS, since it supplies a highly accurate absolute localization [2]. Nevertheless, the use of an RTK-GPS is still quite expensive and is not always reliable, since GNSS signals are disturbed close to building [6], and cannot be used inside greenhouses. Moreover, when working on a field, the task is related to the vegetation and not necessarily to an absolute position.

As a result, alternative solutions may be used, based on exteroceptive sensors such as vision [7] [8]. Nevertheless, such approaches rely on lightning conditions, which cannot be controlled in the different applications of agriculture, especially in open fields. Lidar then raises as an interesting solution, as it is less sensitive to light conditions, while remaining relatively cheap. Several algorithms have been developped, especially based on simultaneous localization and mapping [4] [11] or occupation grid [9]. This implies the use of a map, requiring the detection of specific landmarks [5], which cannot be necessarily visible, because of vegetation changes.

In this paper a lidar sensor is used in order to detect and follow the footprints let by the wheels during previous field operations. These footprints indeed remain unchanged and detectable regardless of the vegetation growth, and are supposed to be an accurate reference for the robot position. The idea is to take part of the inclination of a lidar settled in front of the robot and inclined with respect to the soil (as depicted on Fig. 1). Such an inclination permits to anticipate row curvature, and to build a smoothed local trajectory.

Fig. 1. Illustration of footprint tracking using lidar

The algorithm proposed in that paper is based on several steps. First, the detection of a footprint is performed in the local framework of a laser scan, using a correction of global shape of the soil. This is achieved thanks to a correlation method, with respect to an expected shape with given properties. This permits to define a point belonging to a global trajectory. This latter is then built by the aggregation of such points, obtained by the detection achieved in the successive lidar scans. This accumulation of points is based on the robot odometry, the drift of which is not significant, as only a local trajectory is required. Thanks to these points, a local trajectory is computed on-line in order to obtain all the required variables to feed a simple path tracking control law. Experimental results show the efficiency of the approach through motion in actual field, using footprints on the soil, let by a farm tractor.

This paper is organized as follows. First, the model used for the mobile robot and the structure detection principle are

[1] Irstea, Technologies and Information Support System Research Unit, 9 avenue Blaise Pascal, CS 20085, 63178 Aubière, France firstname.lastname@irstea.fr

[2] SITIA, 7 rue de l'halbrane, 44340 Bouguenais, France t.solatges@sitia.fr

presented. Then, the algorithm developed to build on-line a trajectory from successive footprints detections is detailed. Experimental results are finally proposed and discussed, allowing to highlight the performances of the proposed approach.

II. MODELING

A. Mobile robot modeling and control

In this paper, the control law used is designed for a car-like mobile robot, which can be classically viewed as a bicycle model. Fig. 2 depicts this point of view and introduces the notations used in this paper. These variables are defined as follows:

- v is the robot linear speed,
- $\dot{\theta}$ is the robot angular speed,
- δ is the robot steering angle,
- y is the lateral deviation which is basically considered as the path tracking error, as long as the desired offset is null,
- $\tilde{\theta}$ is the angular deviation. It is the difference between the robot heading in the global space and the trajectory tangent orientation at the closest point to the control point of the robot belonging to the desired trajectory. The point to be controlled on the robot is the middle of the rear axle,
- L is the robot wheelbase,
- $c(s)$ is the trajectory curvature at curvilinear abscissa s.

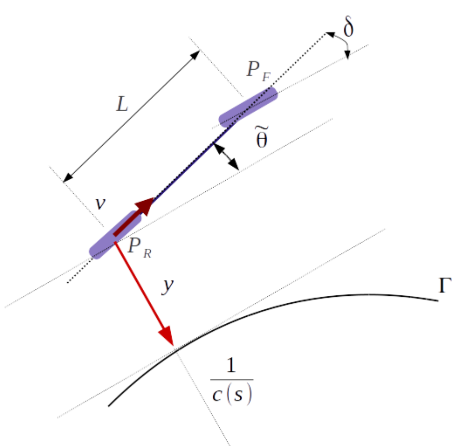

Fig. 2. Simple bicycle model applies to path tracking

This paper does not focus on the path tracking control algorithm, which is ensued from previous works and classical approaches (see [10]). As a result, Eq. (1) expresses the control law applied in this paper, which has been proven to be efficient, as long as assumptions are respected, basically that sliding and velocity are limited. This permits to achieved successive tasks using the same paradigm with several kind of sensors for control (row following, half turn, trajectory tracking to go back to initial position)

$$\begin{aligned}\delta = \arctan(L.(\frac{c(s).cos(\tilde{\theta})}{1-y.c(s)} + \frac{cos^3(\tilde{\theta})}{(1-y.c(s))^2}.\\ (-K_D.(1-y.c(s)).tan(\tilde{\theta}) - K_P.y+\\ c(s).(1-y.c(s)).tan^2(\tilde{\theta}))))\end{aligned} \tag{1}$$

This equation is defined if the condition $1 - y\,c(s) \neq 0$ is true. This means that the center of the curvature should not be superimposed on the center of the robot P_R. The goal of that control law is to bring lateral deviation y and angular deviation $\tilde{\theta}$ to zero. In expression (1), K_P and K_D are tuning gain allowing to defined the robot behavior. We assume that the velocity of the robot will be low enough to ensure that actuators will not be saturated.

$$\dot{\theta} = \frac{v.tan(\delta)}{L} \tag{2}$$

The robot used for experimental results is a skid-steering robot i.e. its control variable is an angular speed instead of a steering angle. As the efficiency of control law (1) is proofed, we use it with an equivalent class defined by Eq. (2) which is also described in [10].

B. Terrain and footprint models in lidar framework

This paper aims at detecting the footprint let by tires in a previous field operation, typically by a farm tractor, thanks to a single-beam lidar, mounting in front of the robot and oriented in soil direction. A desired shape for this footprint is then expected, and the objective of the first step consists in finding a matching shape. For that purpose, a correlation algorithm is used in order to match the current lidar scan and a pre-defined model. This template, as represented on Fig. 3, is representative of one tire footprint, allowing the detection with sufficient accuracy, while limiting the false positive detection result.

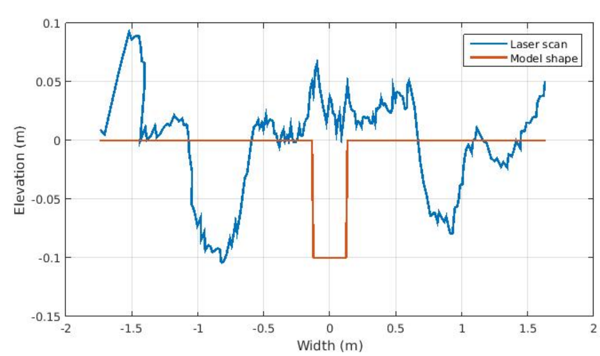

Fig. 3. Example of laser scan with two tire footprints and the associate template use in correlation

We assume that the model and the laser scan have the same number of points. The correlation expression used is defined by :

$$r(k) = \frac{\sum_{j=1}^{N}(z_{j+k} - \bar{z}).(m_j - \bar{m})}{\sum_{i=1}^{N}\sqrt{(z_i - \bar{z}).(m_i - \bar{m})}}. \tag{3}$$

Where :

- ◇ N is the total number of points of both lidar data and template,
- ◇ $r(k)$ is the correlation score for the offset k between laser data and template; with k moving from $-\frac{2N-1}{2}$ to $\frac{2N-1}{2}$,
- ◇ z values are the lidar scan data,
- ◇ m values are the model data,
- ◇ $\bar{z}$ and $\bar{m}$ are the mean value of laser and model data respectively.

The correlation score r must reach a maximum when laser data delayed by k are closest to the template. It is assumed that for k varying from $-\frac{2N-1}{2}$ to $\frac{2N-1}{2}$, it exists a value k_{max}, for which r reaches a maximum r_{max}. Nevertheless, the challenge is to be sure that this maximal value r_{max} makes sense, and can be considered as a confident detection of the expected template. This problem is addressed in the next part of this paper.

III. STRUCTURE RECONSTRUCTION

The goal of this application is to follow a trajectory defined by a structure in the soil with an autonomous mobile robot, thanks to a lidar detection. The approach then consists in detecting the expected structure in each laser scan, and aggregating the successive scans in a global map. The accumulation of points in the 3D space during robot motion will then defined the trajectory to follow.

A. LIDAR scan processing

The first step is to detect the templates defined previously in each lidar scan. Fig. 3 defines the expected footprint in the framework of laser scan and suggests the use of correlation approach, as expressed by Eq. (3). However, in an off-road context, the terrain may quickly appear to be uneven, unlike the expected template. It leads to estimation error or false detection. As a result, a preliminary step consists in estimating the general terrain profile in order to compensate for soil inclination. This is achieved via a least mean square approximation. Data are approximated to a third degree polynomial expression as in Fig. 4. The polynomial degree is chosen in a way that it removes large variations of ground, but does not minimize other variations, which are representative of the template structure. The typical result of this terrain compensation is depicted in Fig. 4.

Once the general shape of the soil is compensated, the correlation equation (3) is used. As pointed out previously, the aim is to recognize the expected template in the lidar scan. In Fig. 3, the model is here defined as a negative hole in the ground with a depth and a width of ten centimeters. This is achieved by applying the correlation Eq. (3), with a varying offset k. The best position for model in data is represented by the maximum of correlation score $r(k)$, such as depicted on Fig. 5. As the Eq. (3) is normalized, a

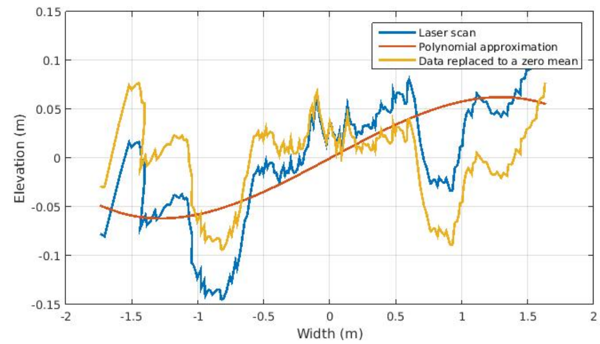

Fig. 4. Removing ground variations

threshold can be chosen in order to validate that the maximal value $r_{max} = r(k_{max})$ is representative of a good detection. This threshold has been experimentally set up to 0.2, as it limits the false positive detection rate while maintaining useful results. As it will be presented in the experimental results part, this threshold is particularly important when using a template composed of multiple footprints in one lidar scan. In that case, not only the maximum value is kept as a good position but all the local maxima are considered as a possibility for structure position.

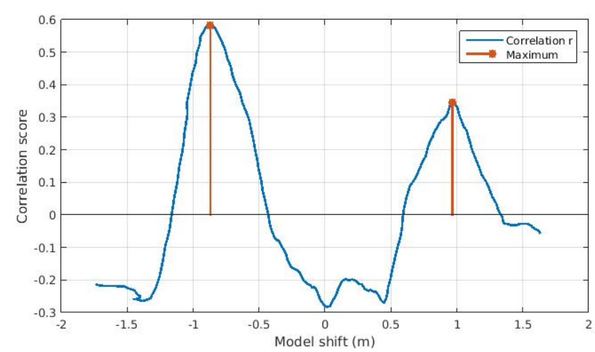

Fig. 5. Result of correlation computation on the data depicted on Fig. 3

The maximum correlation score value gives the shift k_{max} between template position (in the example, two maximum are found as there is two tire footprints in the data) and structure in real data. Therefore, as the three dimensional position of each laser point is known in a global map with a coordinate transformation between lidar and robot, the position of the footprint can be extracted from a reference map.

B. Global map reconstruction

With the previous part of this paper, the structure position can be extracted at each lidar scan and gives a lateral tracking deviation. In order to be stable, the tracking control law (1) requires an angular deviation which could be compute only with several points. The next step then consists in aggregating data with the mobile robot motion in order to define a trajectory to follow in a global map. By means of this data collection, aberrant information can be removed. The main problem about aggregation lies in the accuracy of the robot motion estimation in the global map between each laser scan. In order to avoid the use of expensive sensors, the robot odometry is used for this step. The drift in the use of this kind of data will not affect the accuracy of path tracking, since only a local part will be used to evaluate lateral and angular deviations. In fact, only data between lidar impact and robot tires are kept; this distance of a few meters is not significantly impacted by odometry drift.

Since the footprints are expected to be realized by a previous vehicle in row, one can assume that desired trajectory can be approximated by a local line. It is then possible to aggregate the hole detected at each lidar scan iteration in a continuous line with a Least Mean Square algorithm. The Fig. 6 shows this approximation at three different moments. In the real time application, the linear estimation is made at each iteration of control.

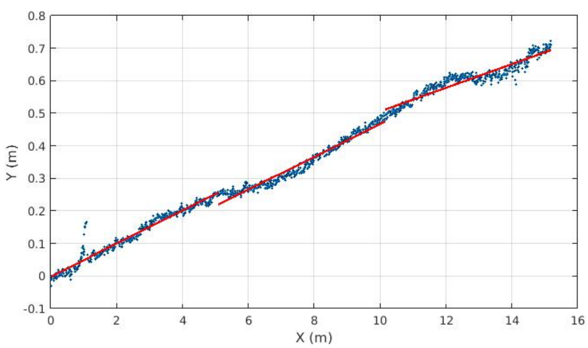

Fig. 6. Linear approximation (red lines) of accumulated structure positioning data (in blue) for the first, second and third part of data

One can take part of the aggregation step to check the validity of the template detection in each laser scan. It has indeed been assumed that the trajectory to be estimated should be continuous. As a result, if the optimal k_{max} is discontinuous, one can deduce that this previous estimation is not relevant.

The last step consists in computing a steering control for the mobile robot. As shown in Eq. (1), it is necessary to extract a lateral and angular deviation between mobile robot and the structure to follow. If the linear estimation of trajectory gives an expression $y = a.x + b$, the gaps can be computed from:

$$y = \frac{a.X_{robot} - Y_{robot} + b}{\sqrt{a^2+1}}, \tag{4}$$

$$\tilde{\theta} = \theta_{robot} - tan^{-1}(a). \tag{5}$$

With this algorithm, the computed control is always updated regarding both motion of the robot and new structure detection.

However, a limitation appears with this method: a minimum number of distant points is necessary to approximate a line and so to compute a relevant control. That means the start of the robot is made in a "blind" process if another positioning system is not reachable. Data acquisition on about two meters length seems to be the minimum value to compute a reliable control. That is why the lidar position and orientation on the robot are important, because it permits seeing further than the robot front, and it also avoids laser masking.

IV. EXPERIMENTAL RESULTS

For the field experiments, we use a skid-steering mobile robot presented on Fig. 1, whose features are presented in Table I. This robot has two driven wheels at the rear and two caster wheels at the front. In this application, there are two control inputs: the linear speed and the angular speed.

TABLE I
ROBOT FEATURES

Weight (with cultivator)	$2100\,kg$
Wheel track	$1.40\,m\,to\,1.80\,m$
Maximum angular speed	$2\,rad.s^{-1}$
Maximum speed	$1.5\,m.s^{-1}$

For the detection process, the chosen LIDAR is an outdoor LIDAR Sick LMS151. All features are presented in Table II. In this application, only 120° in the center of view are kept. In fact, other data are not useful because the ground is not present on it so they are not reliable for the detection algorithm and it would just take more computation time for a useless result.

TABLE II
LIDAR FEATURES

Designation	$LMS151$
Maximum aperture angle	$270°$
Rotation frequency	$50\ Hz$
Angular resolution	0.5○
Operating Range	$0.5\ to\ 20\ m$

All algorithms are implemented on ROS (Robot Operating System); several nodes which compute all actions presented in this paper are used:

- ◇ A mobile robot control node which is publishing the odometry data and subscribing to the control message (steering angle and constant linear speed).
- ◇ A detection node which is subscribing to the LIDAR data and is publishing the position of detected structure.
- ◇ A control node which extracts the control from the lateral and angular deviation between robot and structure positions.

In the next part of this paper, we will show some experiments made in a real world context. First experiment aims at validating the localization based on lidar. The second part shows the effective control of the robot inside a greenhouse for weeds removing.

A. *Detection accuracy: GPS validation*

To demonstrate the accuracy of the algorithm, we use a RTK GPS which has a centimetric accuracy. The experiments presented in this part are conducted in an open field to ensure that a fix GPS measurement is always available. The tracked tire marks are made with the robot itself with GPS position recording. Then, the footprint detection algorithm is used to follow it, always with GPS recording. As GPS and footprints detection does not work in the same coordinates (global or local), we choose the lateral deviation as precision metric which is not coordinate system dependent.

In experiments displayed here, the robot begins with a lateral deviation of about $0.5\,m$. The initialization distance for laser detection is fixed at $2\,m$.

The comparison of robot positions between the footprints creation and the use of the laser following algorithm is presented Fig. 7. Note that the reference GPS trajectory (in blue) was filtered to smooth it while the other data are not.

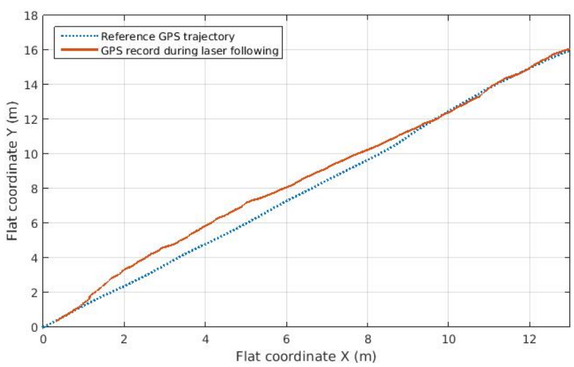

Fig. 7. Result of footprint tracking evaluate with RTK GPS

During the initialization process, the robot is just moved forward without any angular speed. Consequently, it first goes away from reference trajectory and comes back to it when the command process takes over.

As mentioned earlier, the lateral deviation is a good criteria for algorithm validation in comparison with GPS. In fact, it will also be the only available metric without GPS (like in greenhouse) to conclude on algorithm efficiency. In Fig. 8, lateral deviation obtained both by GPS and lidar detection during the same experiment as before are presented.

The result shows that both plot have the same shape even if we can note some gap of the order of $5\,cm$. Actually, this value can be defined as the precision of our algorithm taking into account the relative error of RTK GPS. Then we can consider that the lateral deviation given by lidar detection algorithm is relevant to qualify the footprints following.

In Fig. 8, we can also observe that the robot takes around $10\,m$ to get back on footprints which is coherent with the command law control gain used in this experiment : K_d is

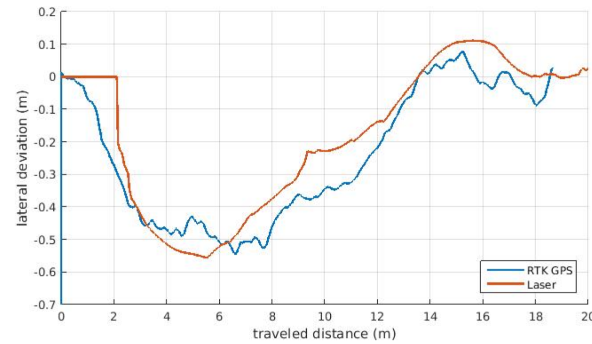

Fig. 8. Lateral deviation compare between lidar detection algorithm and RTK GPS

equal to 0.7 and $K_p = \frac{K_d^2}{4}$. This value could be decreased but that could bring some overshoots or oscillations which are not wanted during the final application in greenhouse.

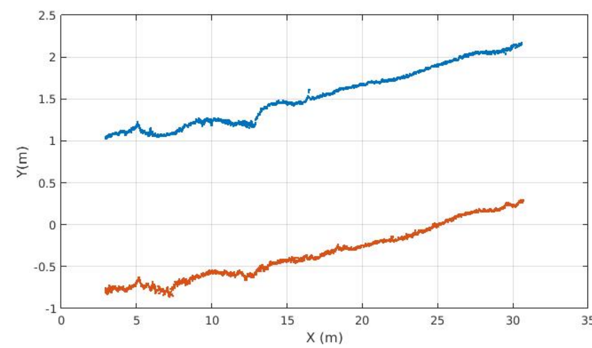

Fig. 9. Reconstruction result with two parallel slots

The last thing on detection validation is to know if the algorithm gives us noisy points or not. It would define the robustness of our application. Fig. 9 shows the points found by the algorithm during the experiment. As presented before, local positioning use only the last $2.5\,m$. In this plot, we can notice that there are no holes in detection process and no false positive points. The distance between two reconstructed lines is also the same as the wheel track of the robot.

At last, we can see that points are close to a line which confirms that the linear approximation used for local positioning is relevant.

B. *Greenhouse experiments: final application validation*

As presented in the introduction to this paper, the goal of this work is to mechanically remove weeds with an autonomous robot in greenhouses. The previous part gave proof of the algorithm precision in an open field; in this part we will present some results obtained in a $40\,m$ long greenhouse.

The experiments were carried out for different values of the robot speed, along with whether the cultivator is used or not. This is an important part as it allows us to test our algorithm in real working conditions.

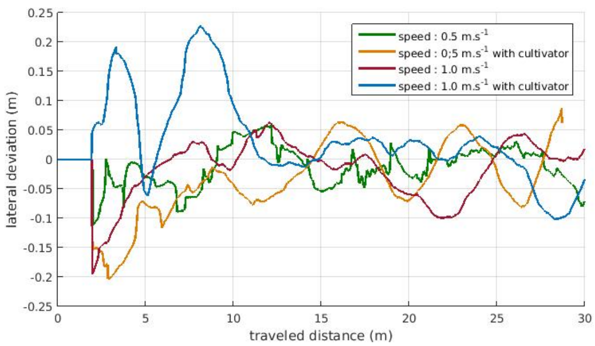

Fig. 10. Lateral deviation at different speed and with or without cultivator

We choose two different speeds: $0.5\,m.s^{-1}$ and $1.0\,m.s^{-1}$. These values represent the actual speed limits used to work with a manual farm tractor. The cultivator used is also an actual tool for pulling weeds. It is important to do some experiment with it because it could change the robot reaction and so our algorithm answer.

In Fig. 10, lateral deviations of four representative experiments are plotted. We cannot observe differences between each tests: in all cases the robot stays with a lateral deviation close to zero after the settling distance of $10\,m$. In these tests, the robot traveled around $30\,m$ autonomously. This distance allows us to confirm the algorithm stability and that whether the cultivator is used or not is not significant for the result.

We have also done some repeatability tests at $0.5\,m.s^{-1}$. After eight experiences on different parts of the greenhouse, which represent a cumulated traveled distance of around $250\,m$, we found an average lateral deviation of $2\,cm$ with a standard deviation of $7\,cm$. Since we do not observe precision variations, these values confirm that our algorithm is efficient when used in working conditions.

V. CONCLUSIONS AND FUTURE WORKS

In this paper, we present an approach to detect footprints and follow it with a mobile robot in real time operation with a 2D laser telemeter sensor. The proposed method is based on a local mapping which does not need a global positioning system or a previous mapping operation.

The method and experiments presented here show that the idea of using a single low cost lidar can be a relevant solution to detect and follow footprints or little ground structures. Accumulating data with mobile robot moves allows reducing positioning errors due to vehicle odometry use. As mentioned in the paper, the main disadvantage of this method is its beginning. In fact, this algorithm requires a first movement of the robot to give a relevant position in local map. This problem could be resolved with a global autonomous approach, for example other positioning systems can be used to bring the robot at greenhouse and to make a U-turn. The robot could be positioned with another solution during the laser initialization distance so there is no more blind process.

The perspectives for this work can address other problems like prediction, detection and estimation of sliding, adaptive control or obstacles avoidance. The method of detection presented in this paper could be adapted to extract more information from lidar measurements. In the same way, more efficient control law could be used or developed to improve mobile robot navigation efficiency in agricultural robotics. This application can also be adapted to differents templates; that could permit to follow other shapes, such as structured vegetation.

VI. ACKNOWLEDGEMENTS

This work is related to the PUMAGRI project which is funded as part of the FUI 20 by Economy, Industry and Digital French Minister and the French region of Pays de Loire and Auvergne.

REFERENCES

[1] Marcel Bergerman, John Billingsley, John Reid, and Eldert van Henten. Robotics in Agriculture and Forestry. In *Springer Handbook of Robotics*, pages 1463–1492. Springer International Publishing, 2016.

[2] D.M. Bevly, J. Ryu, and J.C. Gerdes. Integrating INS Sensors With GPS Measurements for Continuous Estimation of Vehicle Sideslip, Roll, and Tire Cornering Stiffness. *IEEE*, 7(4):483 –493, December 2006.

[3] Simon Blackmore. Towards robotic agriculture. In *SPIE Commercial+ Scientific Sensing and Imaging*, pages 986603–986603. International Society for Optics and Photonics, 2016.

[4] Guillaume Bresson, Thomas Fraud, Romuald Aufrre, Paul Checchin, and Roland Chapuis. Real-time monocular slam with low memory requirements. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):1827–1839, 2015.

[5] ZJ Chong, Baoxing Qin, Tirthankar Bandyopadhyay, Marcelo H Ang, Emilio Frazzoli, and Daniela Rus. Synthetic 2d lidar for precise vehicle localization in 3d urban environment. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1554–1559. IEEE, 2013.

[6] Tomislav Kos, Ivan Markezic, and Josip Pokrajcic. Effects of multipath reception on gps positioning performance. In *Elmar, 2010 Proceedings*, pages 399–402. IEEE, 2010.

[7] Thomas Lemaire, Cyrille Berger, Il-Kyun Jung, and Simon Lacroix. Vision-based slam: Stereo and monocular approaches. *International Journal of Computer Vision*, 74(3):343–364, 2007.

[8] Peyman Moghadam, Wijerupage Sardha Wijesoma, and Dong Jun Feng. Improving path planning and mapping based on stereo vision and lidar. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, pages 384–389. IEEE, 2008.

[9] Julien Moras, Véronique Cherfaoui, and Philippe Bonnifait. Credibilist occupancy grids for vehicle perception in dynamic environments. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 84–89. IEEE, 2011.

[10] Claude Samson, Pascal Morin, and Roland Lenain. Modeling and control of wheeled mobile robots. In *Springer Handbook of Robotics*, pages 1235–1266. Springer, 2016.

[11] Guillaume Trehard, Zayed Alsayed, Evangeline Pollard, Benazouz Bradai, and Fawzi Nashashibi. Credibilist simultaneous localization and mapping with a lidar. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 2699–2706. IEEE, 2014.