Advances in Animal Biosciences: Precision Agriculture (ECPA) 2017, (2017), **8:2**, pp 183–188 © The Animal Consortium 2017
doi:10.1017/S2040470017001157

advances in animal biosciences

Robotic real-time 3D object reconstruction using multiple laser range finders

P. Lepej[1], M. Lakota[2] and J. Rakun[2†]

[1]*Vistion d.o.o., Ljubljana, Slovenia;* [2]*Faculty of Agriculture and Life Sciences, University of Maribor, Slovenia*

An accurate 3D model of an outdoor scene can be used in many different scenarios of precision agriculture, for instance to analyse the silhouette of a tree crown canopy for precision spraying, to count fruit for fruit yield prediction or to simply navigate a vehicle between the plant rows. Instead of using stereovision, limited by the problems of different light intensities, or by using expensive multi-channel 3D range finder (LIDAR scanner), limited by the number of channels, this work investigates the possibility of using two single channel LIDAR scanners mounted on a small robot to allow a real-time 3D object reconstruction of the robot environment. The approach used readings captured by two LIDAR scanners, SICK LMS111 and SICK TiM310, where the first one was scanning horizontally and the second one vertically. In order to correctly map the 3D points of the readings from the vertical sensor into a 3D space, a custom SLAM algorithm based on image registration techniques was used to calculate the new positions of the robot. The approach was tested in an indoor and outdoor environment, proving its accuracy with an error rate of 0.02 m ± 0.02 m for vertical and −0.01 m ± 0.13 m for the horizontal plane.

Keywords: SLAM, LIDAR, 3D reconstruction, field robot

Introduction

A 3D map of objects of an environment is useful in many issues of precision agriculture. For instance, it could be used to localise a robot on the field, to prevent it from bumping into obstacles or to plan a trajectory the robot would take to reach its destination (Roh *et al.*, 2013). Next, a 3D model could be used to get information about the plant development (Garrido *et al.*, 2015), to evaluate the shape / density of a tree silhouette for precision spraying application (Wellington, 2012). And finally, a 3D model could even be used to pick fruits (Dzitac *et al.*, 2012), to measure the size of the trees (Ehlert *et al.*, 2009) or to distinguish weed from maize plants (Andujar *et al.*, 2013).

To make an autonomous field robot it has to include all necessary sensors and logic in order to localise itself, to build a map of the environment and to possibly plan a path it would take to move to a new position. The environment sensing can be accomplished by using simple ultrasonic or infra-red sensors that produce trivial readings, but if we want to get more accurate results, complex sensors should be used, such as cameras, depth cameras, or single/multi-channel LIDAR scanners that produce more detailed results.

Some limitations can be overlooked if we know that the robot will only have to avoid obstacles that are on the ground, where a simple 2D LIDAR can be used to drive between them. In more complex environments, where parts of the objects hang and obstruct the robot, there are more problems. In this cases it is necessary to build a 3D model of the scene. An example of such approach was described by Ben-Tzvi *et al.* (2010), where two cameras were used to calculate a depth information by analysing corresponding pixel pairs. The work from Jung *et al.* (2014) is based on the use of digital camera and a projector that emits structured light on to the target and depth information is recorded by measuring the intensity of the reflected light. The closer the object is, the bright the colour and vice versa. The most expensive sensors are the 3D LIDAR scanners (Roh *et al.*, 2013). They use laser light to measure the distance from the scanner to a point of the scene. In comparison to the ordinary 2D models, the 3D versions have multiple channels. This means they take measurements on different planes, but are limited with the number of channels, for instance 16, 32 or 64.

If a 3D LIDAR scanner is used on a field robot it would make possible for such a robot to sense the environment and navigate between the obstacles (Roh *et al.*, 2013). It would be able to detect the crop lines when placed in a field and navigate between them without destroying the plants. Next, it could detect fruits or crops with an intent to harvest them by focusing on the shapes measured by a 3D depth sensor (Dzitac *et al.*, 2012), or it could record the shape of the tree crown canopy (Wellington, 2012) to adjust the dosage rate of pesticides based on the thickness of the tree crown. If there are

† E-mail: jurij.rakun@um.si

https://doi.org/10.1017/S2040470017001157 Published online by Cambridge University Press

no branches, the valves are shut, with branches the valves are open and if there as just a few branches, the valves are partly open.

It is useful to know the size of fruit or plants. If a fruit producer wants to make a prognosis about the yield of the harvest (Stajnko *et al.*, 2004), he has to know the number and size of fruit. If a forester wants to estimate the biomass (Ehlert *et al.,* 2009) of a tree it has to know its high and diameter or a volume. In all these cases having a 3D model of the scene would be useful and the system like this could be a part of an autonomous farming robot.

One of the most important parts of the software for such a robot would represent a SLAM (simultaneous localisation and mapping) algorithm. The most well-known SLAM algorithms are Gmapping (Grisetti *et al.*, 2005 and 2007) and Hector mapping (Kholbrecker *et al.*, 2013). To localize and create a map of the environment Gmapping relays on the use of Rao-Blackwellized particle filter that minimizes the number of particles to reduce local uncertainty. The approach is optimised for long-range data captured by a LIDAR scanner and therefor suitable for large areas. But in order to work it needs additional data from the robots' odometry.

As tested in a previous work by Lepej *et al.* (2016), Hector mapping is not proven to be the best choice for outdoor environments due to changing conditions and unstructured pattern of the surrounding objects. Instead, an image analysis-based approach is suggested because it utilises image registration techniques done in the frequency domain. In comparison to Gmapping the presented approach works without any odometry data.

The goal of this work is to investigate the possibility of using two non-costly 2D LIDAR scanners to reconstruct a 3D scene, instead of using a more expensive multi-channel sensor, by applying an alternative SLAM approach. To test the approach a small field robot was fitted with two 2D sensors and the tests were conducted in indoor as well as outdoor environment.

Materials and methods

To capture the data needed to build a 3D representation of the world, a small autonomous field robot was used in this experiment. The robot is originally mounted with one LIDAR scanner (SICK TIM310) that enables it to move between the crop lines, but to build a 3D model an additional sensor (SICK LMS111) was mounted to record shapes on vertical plane.

Then to build a 3D model of the scene a two-step approach was used. The first step represents the mapping algorithm (SLAM) that builds a 2D map of the environment. It is then followed by the second step, point cloud assembly, which adds additional dimension to the first.

Instrumentation - Cornstar robot

To test the approach a small robot was used, build as part of a student reserch project. The robot has four components: sensors, embedded electronics, mehanics and actuators. qThe mahanical part of the robot is build as a four-wheel drive, where the two axles can turn indivitualy, a three phase motor that enables the robot to move and automatic gearbox with one backward and two forward speed transition. To spray the robot also has two electo-magnetic valves, two nozzles and a preassured reservoir. The sensory part of the robot includes the LIDAR scanners and one on-board digital camera. All meassurements on the robot are captured by an embedded computer with ROS and LINUX installed on it. It job is to process the data and make new decisions in order to autonomously drive the whole platform. The robot is depicted on Figure 1.

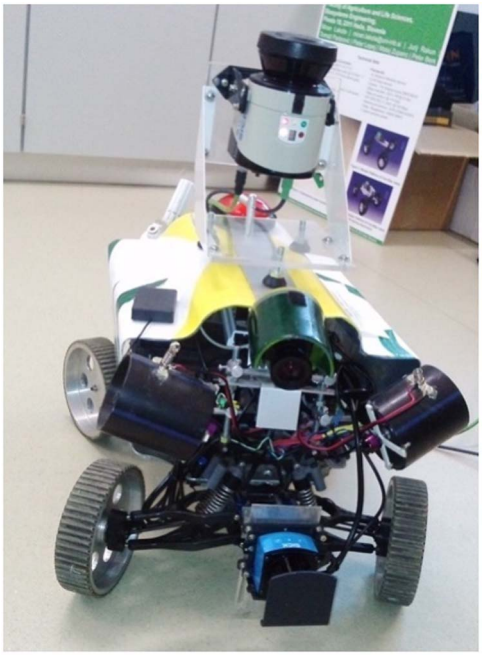


Figure 1 Cornstar robot with two mounted LIDAR scanners. The LMS111 is mounted horizontally and TIM310 vertically.

Methods – SLAM

In order to build a 2D map of the environment and localise the robot on a plane the SLAM algorithm takes measurements from the horizontally mounted LIDAR scanner (Grasetti *et al.*, 2007; Kohlbrecker *et al.*, 2013; Lepej *et al.*, 2016). Based on the relative position of the robot a static transformation into 3D space for the data measured by the second, vertical LIDAR scanner is made. By doing so and combining multiple readings in to the same space a 3D model of the environment is build. To achieve this 3D model assembler in ROS and point cloud library were used to construct a 3D model of the world.

Methods – SLAM - Phase Correlation

To calculate displacement between each consecutive measurement, phase correlation was chosen as described in (Gonzalez & Woods, 2001) and (Stone, 2001). The correlation works by comparing two images, where the first is a reference and the subsequent one is a template. Since the measurements are semi-identical, looking from a signal point of view, they have almost the same energy distribution.

https://doi.org/10.1017/S2040470017001157 Published online by Cambridge University Press

This means the correlation reaches a maximum value when both images are registered. In addition, if the correlation is computed in frequency domain, the translation and rotation are separated and presented in the phase (translation) and amplitude (rotation) spectrum, without having an effect on to another, as in spatial space. So the problem of finding the right parameters can be solved separately.

The translation of the semi-identical signals in spatial space only affects the phase spectrum of the frequency transform. As the signals are semi-identical their phase compensates but preserves their linear part. This produces a maximum global value with coordinates (c_x, c_y) that corresponds to the translation on the X and Y axes. If the log-polar transform is used before phase correlation, phase correlation could also be used to calculate rotation. To calculate the correlation, the following equation is applied:

$$I_1(x,y)^{\circ} I_2(x,y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I_1^*(m,n) I_2(x+m, y+n) \tag{1}$$

where $I_1(x,y)$ and $I_2(x,y)$ represent a 2D signal (image) with M x N pixels in size and I_1^* a complex conjugate of I_1. By applying the convolution theorem, Eq. 1 can be written as:

$$I_1(x,y)^{\circ} I_2(x,y) \Leftrightarrow S_1^*(u,v)\, S_2(u,v) \tag{2}$$

where S_1^* and S_2 correspond to frequency transforms of the input signals I_1 and I_2. Based on the convolution theorem the correlation in frequency domain can simply be calculated by multiplying the complex conjugate of the frequency transform of the first by the frequency transform of the second signal.

By transforming measurements and depicting them in spatial space, the consecutive reading shows up as translated outlines of the objects in the scene. Once these displacements are calculated with the help of Eq. 2, they are used to calculate real displacements on both sides. Both displacements are then simply used to compute new vectors with an origin starting at (0,0) located at the beginning of the row, approximately half the distance from each side of the row. Both vectors are then used in simple vector summation which reveals the true movement vector that is the difference along the X and Y axes, along with the new coordinates of the origin for new iteration.

Methods – SLAM – Data Filtering

In real environments the distance between the crop rows may vary, and if the robot is located too close to one side, it can pick up distance readings from the neighbouring row. This can be eliminated in some cases by setting the maximum valid distance reading in ROS's SICK laser finder node. However, in order to eliminate it completely, a preprocessing filtering step is recommended. A statistical approach was chosen to eliminate this kind of reading. By calculating a median distance for each crop row an interval of valid readings was computed. It comprised of 66.66% of all readings that were closest to the median. Since the majority of the distances in all readings belong to the first row, the marginal distances were filtered out.

Methods – SLAM – Estimating pose and orientation

Different crop plant rows can have versatile structures that produce very different environmental conditions. Our approach to this problem is based on image registration, where we chose phase correlation. It computes the information about the template position on a corresponding original image. Our goal was to estimate an angular shift in successive measurements that corresponds to the differences in translation and rotation on images. We have split these two problems, where we need to calculate phase correlation separately for the estimation of position and orientation. Therefore, we need two different data scan transformations between two distinct successive readings. First, we will describe laser data transformation, which is essential for the phase correlation that is used to estimate the pose.

Methods – SLAM – LIDAR scanner data transformation

This transformation adds an additional dimension to the horizontal laser scan data. The first transformation from LIDAR scanner scans to a 2D signal (image) is done, which is actually a projection of laser scan data into a two-dimensional scaled grid map.

Eq. 3 describes the grid map transformation for linear movement estimation. Transformation is performed for every measured distance set captured by the LIDAR scanner, where only valid data is considered that is greater than 0 and smaller than the maximum range of the LIDAR scanner. For each piece of LIDAR scanner data, the x_I and y_I coordinates of the 2D signal are calculated. The same conditions are considered for rotational estimation, where for the rotational estimation the data scan transformation is performed and the whole transformation is described by Eq. 5. Laser data transformation uses threshold limits, which are defined by user settings. Users can define the maximum and the minimum laser range, which should be LIDAR scanner specific and depending on the environment.

$$\begin{bmatrix} x_I \\ y_I \end{bmatrix} = \begin{bmatrix} \frac{round((\cos(\beta)) * Irf(i))}{res} + I_w \\ \frac{round((\sin(\beta)) * Irf(i))}{res} + I_h \end{bmatrix} \tag{3}$$

$$\beta = deg2rad(i + 45) \tag{4}$$

Pose estimation is a linear translation, calculated with the help of a parameter that represents a current laser beam angle. We need to add 45° to obtain the correct orientation of the LIDAR scanner. In our case, we can get a 270 LIDAR scanner measurement, but 180 is used, owing to occluded areas where the wheels of the robot obscure the view. Each measurement, defined by angle i, has a corresponding distance measurement noted with $irf(i)$. The parameter res is the resolution of the signal I with a size of (I_w,I_h). For angular estimation, x_a and y_a signal positions are calculated, where

https://doi.org/10.1017/S2040470017001157 Published online by Cambridge University Press

the parameter $range_{max}$ is used to scale the given LIDAR scanner range into a signal, which has a size two times greater than the LIDAR scanner angular resolution to gain sub-pixel accuracy. The *y* component is presented as angle data.

$$\begin{bmatrix} x_a \\ y_a \end{bmatrix} = \begin{bmatrix} \frac{round(lrf(i) * I_h)}{range_{max}} \\ i \end{bmatrix} \quad (5)$$

By applying Eq. 3 and Eq. 5 to the original data sets produced by the LIDAR scanner, two 2D point clouds are constructed, and two more for the next measured data sets. To determine the difference in rotation, the Eq. 5 was used to produce the other two point cloud maps.

Methods – SLAM – Pose estimation

The first point cloud was used as a reference and part of the other, the successive measurement, as a template. In order to guarantee the same content in both signal, the template was cropped by 100 pixels on each side. Usually the shifts are very small since the laser scanner generates new data rapidly, up to 15 Hz. The result of the correlation is presented as a position on the base signal, which is converted to a signal shift. For a small template shift, we use Eq. 6, as shown:

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} c_x + (W_h - I_w) / 2 \\ c_y + (W_h - I_h) / 2 \end{bmatrix} \quad (6)$$

where W is a template signal that has a size of (W_w, W_h) pixels.

To calculate an angular shift, a similar approach can be used. Instead of converting the reference and a template to log-polar space and then using the phase correlation, the measurements are transformed to (r, θ) space. In this way, the angular shifts show up as translational shifts that are calculated with the help of phase correlation. Eq. 7 shows how the current angular shift is calculated:

$$\theta_c = ((c_y + (W_h - I_h)) / 2), \quad (7)$$

where c_y is the result of phase correlation.

The previous step of phase correlation returned information regarding the shifts on both axes as well as the shift in angle that corresponds to rotation. The translation is then used to determine the movement of the robot, which is calculated by using the following equations:

$$u_x = \sum dx * I_w / 2 \quad (8)$$

$$u_y = \sum dy * I_h / 2 \quad (9)$$

$$\theta = \sum \theta_c \quad (10)$$

The robot or LIDAR scanner pose is represented in (u_x, u_y), where (*dx, dy*) are calculated shifts in two dimensions. Current robot rotation is estimated in θ_c which is given in degrees, and the θ represents the result of current robot orientation.

Results and discussion

In order to prove the approach, two similar tests were conducted; the first in an indoor environment, a corridor, and the second in outdoor environment, an apple orchard, where the conditions change (due to wind). In both cases the reconstructed shapes were measured and compared to real measures for different objects with different sizes; windows for indoor, tree trunks for outdoor environment. In addition, because the vertical readings need to be placed at precise position, the data regarding cumulative length and intermediate distance between scans is considered as well as scene width and height. Table 1 summarizes the results.

Figure 2 depicts the indoor environment and Figure 3 outdoor environment, both with part of the reconstructed 3D scene. Figure 2 clearly depicts objects from the scene and Figure 3 shows a real image of apple orchard and its 3D reconstruction from two slightly different viewpoints. The centre image clearly shows the space between the apple rows, while the right image distinct tree trunks (clusters of voxels in lower right side of an image).

The tests were conducted while the robot was driving at roughly 0.5 m/s with the error rate of 0.015 m ± 0.03 m for

Table 1 *Two different tests (indoor/outdoor environment) depicting real and reconstructed dimensions. Rows marked with (#) correspond to measurements from horizontal (SLAM supported) LIDAR and others to vertically positioned LIDAR*

Test	Description	Real values [m]	Reconstructed values [m]	Avg. error [m]
Indoor environment (corridor)	Width	2.11	2.16	0.05
	Height	2.98	3.00	0.02
	Length(#)	9.16	9.00	−0.16
	Objects - height	1.78	1.81	0.03
	Objects-width(#)	1.25	1.20	0.50
Outdoor environment (orchard)	Width	2.52	2.50	0.02
	Height	3.13	3.15	0.02
	Length(#)	10.0	10.15	0.15
	Objects - height	0.36	0.40	0.04
	Objects-width(#)	0.07	0.09	0.02
		Average absolute error – vertical plane:		0.02 ± 0.02
		Average absolute error – horizontal plane(#):		−0.01 ± 0.13

https://doi.org/10.1017/S2040470017001157 Published online by Cambridge University Press

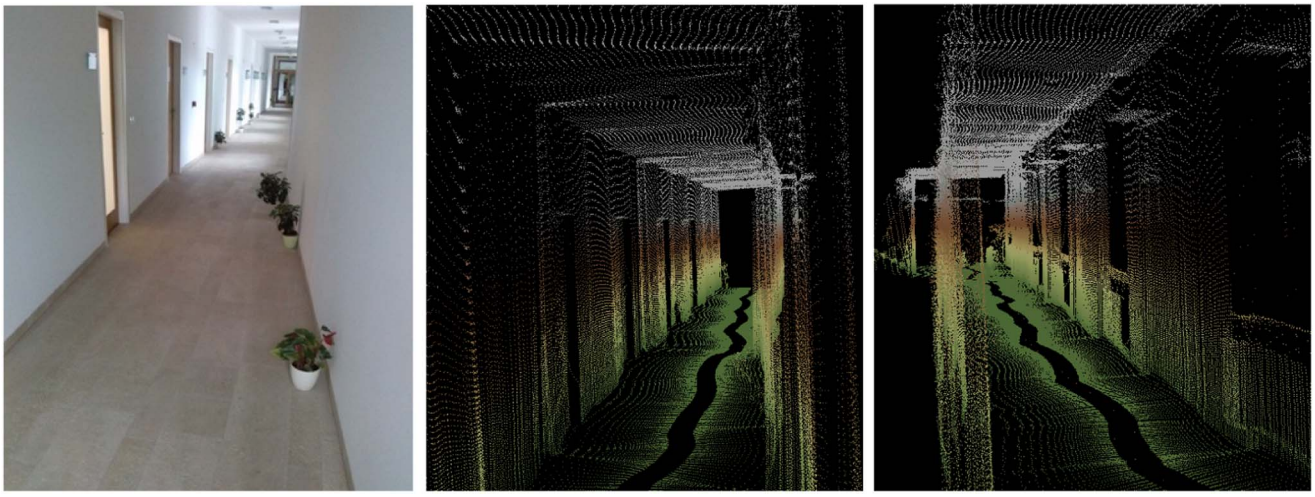


Figure 2 Indoor environment – a corridor. The left image depicts a real scene, while image in the centre and on the right depict their representation in 3D space.

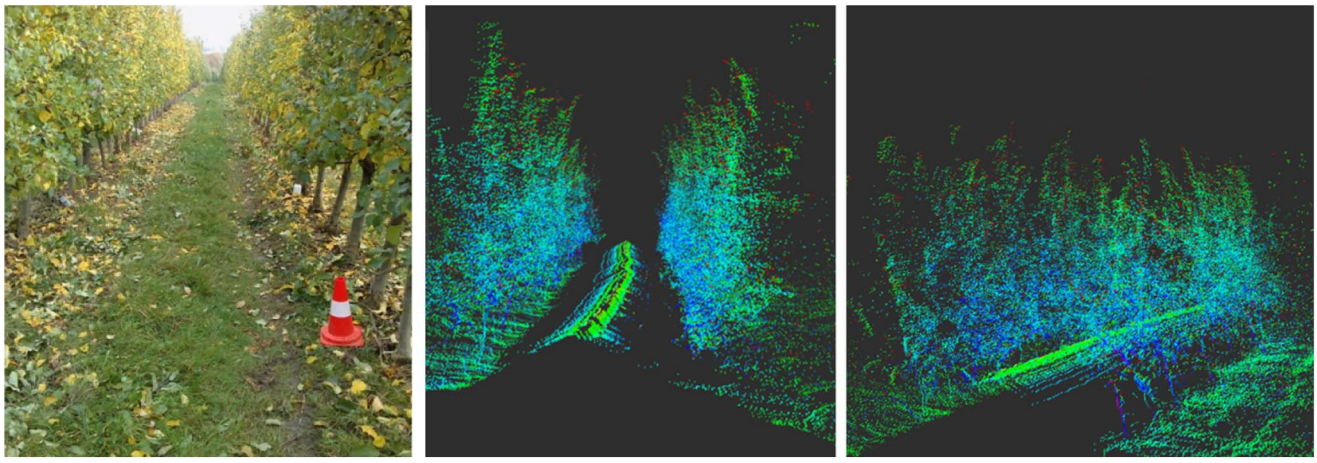


Figure 3 Outdoor environment - apple orchard. The left image depicts a real image, while in the centre and on the right it shows their representation3D reconstruction.

vertical and $-0.03\,\text{m} \pm 0.29\,\text{m}$ for horizontal plane when comparing real to reconstructed dimensions. On average the error rate for estimating metric data in indoor environment was $0.02\,\text{m} \pm 0.02\,\text{m}$ (or $-0.11\,\text{m} \pm 0.08\,\text{m}$ for horizontal and $0.03\,\text{m} \pm 0.015\,\text{m}$ for vertical plane) and $-0.01\,\text{m} \pm 0.13\,\text{m}$ (or $0.09\,\text{m} \pm 0.01\,\text{m}$ for horizontal and $0.01\,\text{m} \pm 0.03\,\text{m}$ for vertical plane). The error rate is a bit higher for horizontal plane as it accumulates over time. The average errors (0.02 m and −0.01 m) are comparable, while the standard deviation is bit higher for outdoor environment due to changing silhouette and the speed the LIDAR is taking measurements vs the traveling speed of the robot.

In setup from Figure 1 the horizontal scanner takes scans at a higher level as the robot height. These cases no problems as far as the SLAM algorithm is concerned. It measures the distance to the walls (indoor) and tree trunks (outdoor). If it is to be used to detect the obstacles in from of the robot as well, the position of the two LIDARs could be switched or the horizontal scanner could be mounted at an angle, for instance at 45°.

Conclusions

This work summarises and evaluates an approach to reconstruct a 3D scene by using two 2D LIDAR sensors mounted vertically and horizontally. Similar results could be achieved by using a multi-channel 3D sensor. The advantages of the proposed solution are that this kind of setup is more economical (around 3- to 4-time less expensive) and it provides a higher degree of detail. The only real drawback is the method how it captures the details, as it has to pass each object instead of detecting it a couple of meters ahead like the multi-channel LIDAR sensor does.

To combine the readings by using two LIDAR sensors the data relative to the robot's position must be available, which can be determined by using a SLAM algorithm. For this we used our own approch that is more efficient than other known solutions when working in outdoor environment where part of the scene could change due to outdoor forces.

The results show that the proposed approach is good enough for some applications, with the error rate of

https://doi.org/10.1017/S2040470017001157 Published online by Cambridge University Press

$0.015\,\text{m} \pm 0.03\,\text{m}$ for vertical and $-0.03\,\text{m} \pm 0.29\,\text{m}$ for horizontal plane. To improve the results LIDAR sensors with higher resolution are suggested and slower robot driving speeds, with possible switch of LIDAR positions to enable the robot to detect obstacles in front of the robot.

References

Andujar D, Rueda-Ayala V, Moreno H, Rosell JR, Escola A, Valero C, Gerhards R and Griepentrog HW 2013. Discriminating crop, weeds and soil surface with a terrestrial LIDAR sensor. Sensors 13, 14662–14675.

Dzitac P and Mazid AM 2012. A depth sensor to control pick-and-place robots for fruit packaging, *12th International Conference on Control Automation Robotics & Vision (ICARCV)*, pp. 949–954.

Garrido M, Paraforos DS, Reiser D, Vazquez Arellano M, Griepentrog HW and Valero C 2015. 3D Maize Plant Reconstruction Based on Georeferenced Overlapping LiDAR Point Clouds. Remote Sensing 7 (12), 17077–17096.

Grisetti G, Stachniss C and Burgard W 2007. Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters. IEEE Transactions in Robotics 23, 34–46.

Grisetti G, Stachniss C and Burgard W 2005. Improving Grid-based SLAM with Rao Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling, Proceedings of the IEEE International Conference on Robotics and Automation.

Gonzalez RC and Woods RE 2001. Digital image processing, 2nd edition, Prentice Hall, Upper Saddle River, New Jersey, USA.

Jung H and Lyou J 2014. 3D map building using the kinect mounted on a mobile robot, IEEE International Conference on Industrial Technology (ICIT), pp. 604–608.

Kohlbrecher S, Meyer J, Graber T, Petersen K, Klingauf U and Stryk O 2013. Hector Open Source Modules for Autonomous Mapping and Navigation with Rescue Robots, Department of Computer Science, TU Darmstadt, Germany.

Lepej P and Rakun J 2016. Simultaneous localisation and mapping in a complex field environment. Biosystems Engineering 150, 160–169.

Roh HC, Sung CH and Chung MJ 2013. Rapid SLAM using simple map representation in indoor environment, 19th Korea-Japan Joint Workshop on Frontiers of Computer Vision, pp. 225–229.

Stajnko D, Lakota M and Hocevar M 2004. Estimation of number and diameter of apple fruits in an orchard during the growing season by thermal imaging. *Computers and* Electronics in Agriculture 42, 31–42.

Stone HS 2001. A Fast Direct Fourier-Based Algorithm for Subpixel Registration of Images. IEEE Transactions on Geoscience and Remote Sensing 39 (10), 2235–2242.

Wellington C, Campoy J, Khot L and Ehsani R 2012. Orchard tree modeling for advanced sprayer control and automatic tree inventory. In: Proc. IROS Workshop on Agricultural Robotics.

https://doi.org/10.1017/S2040470017001157 Published online by Cambridge University Press