

ANALYSIS OF THE SURROUNDING ENVIRONMENT USING AN INNOVATIVE ALGORITHM BASED ON LIDAR DATA ON A MODULAR MOBILE ROBOT

Submitted: 18th September 2019; accepted: 18th March 2020

Nicola Ivan Giannoccaro, Takeshi Nishida

DOI: 10.14313/JAMRIS/4-2020/41

Abstract: *In this paper, a low cost mobile robot with a modular design that permits the easy change of the number of wheels is considered for generation of 3D digital maps by using ROS tools and a 3D light detection and distance measurement (LiDAR) sensor. The modular robot is thought for travelling through several environments with saving the energy by changing the number and arrangement of the wheels according to the environment. The presented robot can construct a 3D map in particular structured environment and the running performance was investigated by an extensive characterization. Furthermore, in light of the experimental tests, a new simple algorithm based exclusively of the processing of the LiDAR data is proposed with the aim of characterizing the surrounding environment with fixed landmarks and mobile targets.*

Finally, the limits of this prototype and of the proposed algorithm have been analyzed, highlighting new improvements in the future perspective development for permitting an autonomous environment perception with a simple, modular and low-cost device.

Keywords: *mobile robot, driving module, 3D digital map, LiDAR, ROS, SLAM*

1. Introduction

In several robotic applications, it is necessary to analyze the surrounding environment to get information about the presence of objects and about the trajectory of the robot respect to landmarks or mobile targets (i.e. the cut trees in forestry where it is necessary to measure the number of trees [1]). Often, in several applications, it is necessary to measure the environment in detail also when it is difficult to identify the position by global navigation satellite system (GNSS) signals; in these cases, the generation of three dimensional digital map by light detection and distance measurement (LiDAR) has been studied [2–4]. Currently, research based on the application of simultaneous localization and mapping (SLAM) by using only LiDAR data is widely conducted. Several methods of measuring the surrounding environment using 2D LiDAR [5, 6] have been proposed applied to four-wheeled mobile robots that installs these algorithms and generate digital maps with real-time

SLAM. In addition, unmanned aerial vehicle (UAV) equipped with LiDAR [7] and measurement by a six-wheeled robot having a rocker bogie mechanism have been developed and their performance has been experimented [8]. On the other hand, it is difficult to predetermine the mechanism and size of mobile robots suitable for various environments and tasks. In each environment, heterogeneity of the ground, aspects of the ground surface, and so on are different. Increasing the contact area by increasing the number of wheels improves the running performance of the robot but reduces the energy efficiency for the movement. Reduction of energy efficiency should be avoided in designing autonomous mobile robots that must carry limited energy sources. Namely, there is a trade-off relationship between the number of wheels and the running energy. If it is a relatively flat ground, a robot can travel sufficiently with an independent two-wheel mechanism and has high energy efficiency. For traveling on a rough meadow, running performance can be improved by running on four or more wheels. In addition, in the case of driving in the forest and sand, a six-wheel rocker bogie mechanism for ensuring runnability is suitable, however its energy efficiency is low [9].

Recent works [10–11] have highlighted that robot operating system (ROS) platform is particularly advantageous for environment mapping generation by using a LiDAR mounted on a mobile robot. Moreover, the possibility of using the LiDAR data (point cloud data) with efficient algorithms for simultaneous robot localization and mapping has also been recently demonstrated (i.e. [12]).

In this research, a simple low-cost robot specially designed, driven by ROS and realized with modularized wheel and a frame for connecting them and for supporting a scanning LiDAR is presented with an extensive characterization of its performance in relation with the LiDAR vibrations respect to the type of surface with which robot is in contact. This is important because the 3D LiDAR sensor mounted on the robot depending on the vibrations acting on the system, can scan the surrounding environment more efficiently with lower oscillations by using recent algorithms introduced in literature. The developed robot, which can be constructed by combining the modules, can adaptively change the number and arrangement of the wheels according to the environment to be measured and by taking into account the results of the present research.

The paper has two further aims:

- 1) To provide an algorithm for tracking the odometry of the mobile robot. This practice is very common in the robotic field and several SLAM algorithms have already been implemented. In most cases, however, the input data of the algorithms are taken from IMU and GPS sensors or, more rarely, from fixed cameras, which inspect a certain control environment. It is clear, though, that both methods have their limitations: in the first case, the application of the method appears to be difficult in shielded environments, due to the impossibility to reach the GPS signal. While in the second case, the equipment needed (such as cameras) would allow the control only over very limited conditions (light, brightness). The algorithm proposed here, on the other hand, uses only the data produced by LiDAR sensor. In fact, LiDAR can provide frames with Point Cloud data which characterize the surrounding environment; hence, the method could become more versatile and able to confront the critical issues described above.
- 2) To link the movement of the proposed modular robot with the surrounding environment, forecasting the kinematics of possible targets and collisions between the moving elements in the examined space.

This algorithm, as a future development, aims to support the research studies related to autonomous driving, a topic of actual and great interest.

2. Design of Robot and Features

2.1. Devices of Robot

In order to control the robot, one laptop computer with Intel Core m5 (1.10 GHz) and Ubuntu 14.04 LTS has been used. In addition, the robot operation system (ROS) [13] was adopted as middleware. Then, the following devices were incorporated into the robot: motor driver (MDD10A) which generates PWM signal, a motor controller (iMCs01) including pulse counter, a 3D LiDAR (YVT-X002, Hokuyo), and a 24 V battery.

The robot system scheme is shown Figure 1. It is possible to control two driving modules with a set of motor driver and motor controller by a signal from the laptop with USB connection. The LiDAR uses power directly from a 24 V battery.

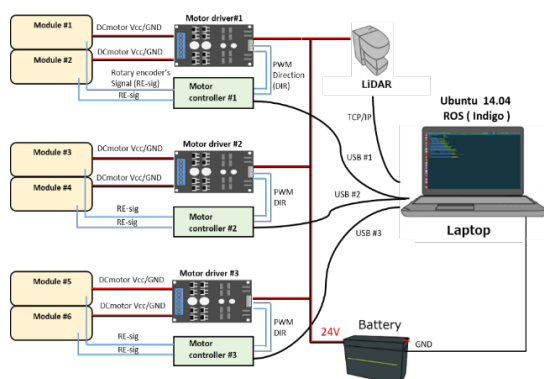


Fig. 1. Signal and electric scheme of the developed robot system

2.2. Frame of the Robot

The frame of the robot is designed so that the number of drive wheels can be changed according to the environment where the robot is traveling. Depending on the environment, the robot can be easily changed by user to energy-efficient wheel versions, six wheel versions with high running ability against complicated terrain, four wheel versions with intermediate capabilities of them or two wheels. Therefore, by designing the frame as an x shape as shown in Figure 2(a), since the width and length of the robot are the same, the robot can execute a spin turn by reverse rotation of each wheel. In the two wheels version, two driving wheels and one slave wheel are used. In the four wheels version, four driving wheels are connected in the same direction to the frame. In addition, for the six wheels version, an attachment frame was developed so that it becomes a rocker bogie mechanism. The proposed driving wheels, the frame, and the attachment frame are shown in Figure 2(b).

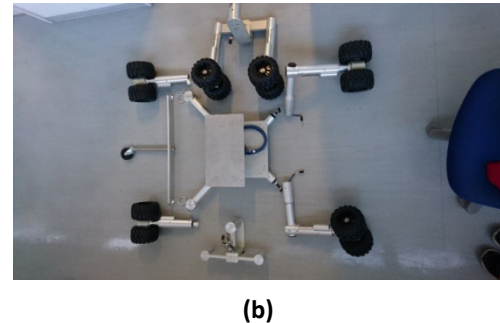
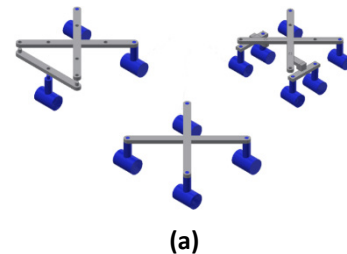


Fig. 2. Combination of the driving modules with developed frames

2.3. Driving Module

The driving module, shown Figure 3, consists of a DC motor (TE-38F16-24-64), a rotary encoder (E6A2-CW3E), and two wheels connected shaft and gears. The total length is 382.5 mm, the total width is 217 mm, and the weight is 2500 g. The wheel's diameter is 150 mm and the maximum velocity is 0.4 m/s. This module has waterproof performance. Also, by equipping two tires in parallel, the driving module became less likely to catch on weeds and branches while the robot was moving. After connecting the connectors of each module and the frame, fix it by screwing with the frame. By measuring the rotation angle of the tire with a rotary encoder, the rotation speed of the tire is controlled. Also, since the sponges are inside the tires, they do not puncture [14].

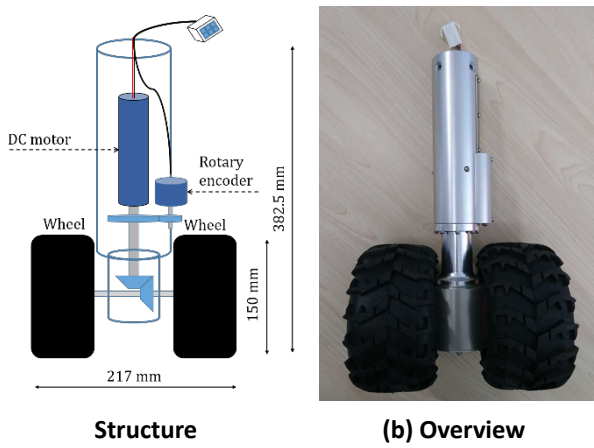


Fig. 3. The driving module consists of a DC motor, a rotary encoder, four gears and two wheels. (a) Structure and (b) over view of a driving module

2.4. Configuration of the Robot

The developed robot can be changed to three different configurations by changing the combination of drive modules as shown in Figure 4. The first type consists of six driving modules and two rocker bogie joint attachment. The second one has four driving wheels, the third type has two driving wheels and one non-driving wheel. Each types are designed depending on an assuming following fields.

- The six-wheel type: this is for traveling across all different conditions of the road (mud, grass, asphalt) and can climb obstacles [15]. This is influenced by the Rocker-bogie mechanism [16-18]. These characteristics are perfect to drive the Robot trough forest environment.
- The four-wheel type: this keeps a great stability during up-hill and cross-hill, because of its symmetrical body, anyway the Robot presents some problems to climb obstacles due to no-bogie joint and no flexible structure.
- The two-wheel type: this is the simplest configuration; Robot has a good stability but risks to tip over the ground if it overcomes obstacles or travels across rough ground.

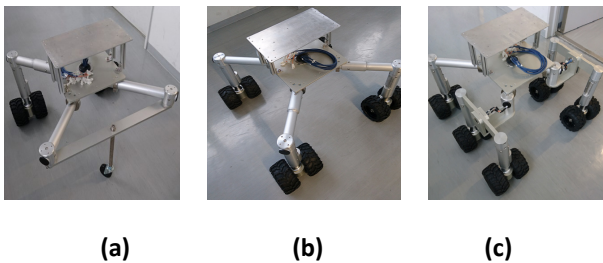


Fig. 4. Three types of the robot configuration with changing drive wheel combinations: (a) two-wheel type with one non-driving wheel, (b) four-wheel type, and (c) six-wheel type with rocker bogie joint attachment

The case anterior part has been designed for receiving and fixing in a correct way the LiDAR sensor, giving it also an adequate protection structure for facing eventual impacts during the robot motion (Fig. 5). The solid angle that is possible to inspect with this sensor model (Hokuyo XVT- 35 LX 3D LiDAR) is quite extensive in both the directions (Fig. 6).



(a)



(b)

Fig. 5. a) Structure for placing the LiDAR sensor, b) The Lidar sensor mounted

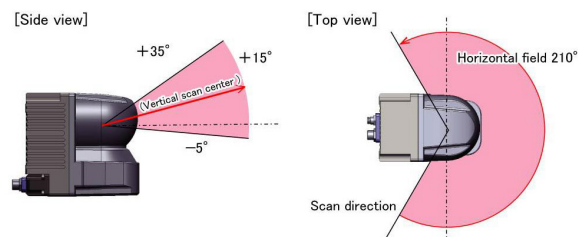


Fig. 6. Serviceable solid angle range of the LiDAR sensor

3. Preliminary Tests

3.1. Relationship Between the Number of Tires and Running Performance

The vibrations induced on the LiDAR sensor, placed on the top of the robot, during its movement could compromise the accuracy of the acquired cloud of points and also could ease the robot organs of unscrewing (as verified during the experiments). For this reason, a reduction of the vibrations choosing the best robot configuration (2, 4 or 6 wheels) could become a strategic factor for the accuracy and the maintenance of the robot. At this proposal, an extensive experimental analysis has been carried out considering the data of the value of the vertical acceleration applied to the LiDAR by means of a Inertial Measurement Unit (IMU) placed inside the LiDAR box.

The tests have been conducted with a first phase of calibration, by considering a preliminary evaluation of the characteristics of the used accelerometer; in the preliminary tests the robot was still on the floor (see Fig.7) and the accelerometer acquired the vertical oscillations for a period of 30 seconds and sampling frequency 50 Hz. The standard deviations σ , expressed by Eq. (1), for 3 different tests are reported in Table 1, demonstrating the similarity of the standard deviation value and giving a confidence value of the vertical accelerometer accuracy.

$$\sigma = \sqrt{\frac{\sum_{i=1}^N a(i) - \bar{a}}{N}} \quad (1)$$

In (1), N is the total number of acceleration samples $a(i)$ and \bar{a} is the mean value, that is considered as accelerometer offset. The parameter introduced in (1) will be considered as the comparison parameter for the vertical oscillation for all the following experimental tests presented.

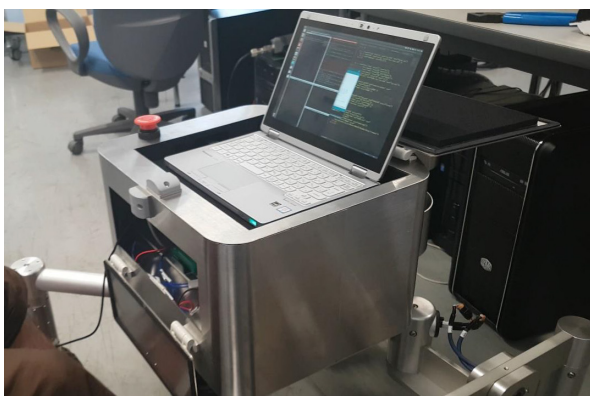


Fig. 7. Setup of the preliminary calibration tests

Tab. 1. Standard deviation of the vertical acceleration for 3 tests with still robot

	First test	Second test	Third test
Vertical acceleration standard deviation	0.138	0.156	0.188

All the possible robot configuration, in relation with the possible surfaces scenarios have been measured when the robot was running at a constant speed. When the acceleration is small, it indicates that the plurality of wheels are appropriately grounded to the environment. If the acceleration is large, the wheels do not properly contact the ground, and a jump accompanying the rotation of the wheels occurs. That is, by comparing the acceleration in the vertical direction for each type of robot, it is possible to evaluate the running performance against the running environment. The values of a vertical acceleration sensor installed in the LiDAR at that time by running the three types of the robot with three types of environment of grassland and gravel road at a constant speed were measured. The three environment are shown in Fig. 8.

In each environments at least 3 experiments for each robot configuration (2, 4 or 6 wheels) have been carried out in order to characterize the robot behavior and also to be guaranteed about the repeatability characteristics and a sampling frequency of 50 Hz has been used. The robot has been drive by a joystick at its maximum speed in the longitudinal direction.

In Fig. 9 an example of the results referred to the environment b) in Fig. 8, asphalt; it is evident a heavy difference of vertical acceleration for the 3 considered configurations. A more clear view may be carried out by Tables 2–4, where for each type of environment, asphalt in Table 2, grassland in Table 3, Gravel in Table 4 the standard deviation of 3 different tests, named Test1, Test2 and Test 3, for each configuration (2, 4 or 6 wheels) are reported. For the tests on gravel surface the configuration with 2 wheels was not able to move the robot, so only the results (2 tests) with others configurations have been reported.

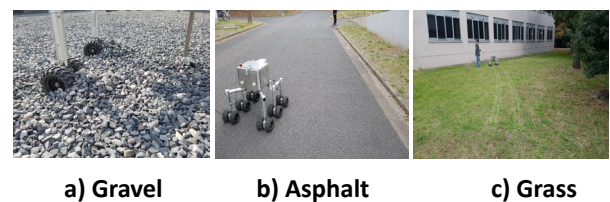


Fig. 8. Three environments for experiments

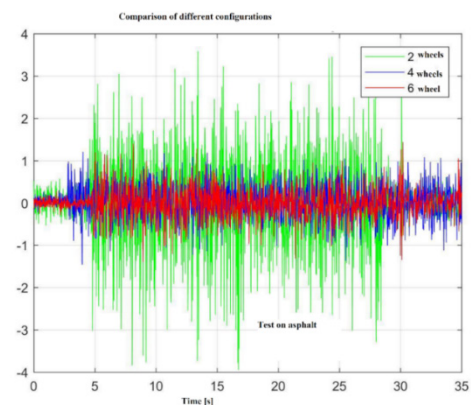


Fig. 9. Vertical acceleration (in m/s^2) of Test 1 on the asphalt for 3 configurations without gravity

Several considerations may be carried out analyzing the results in Tables 2–4 and considering several other experiments that have been conducted also on mixed surfaces or hill-profile and so on:

- The parameter σ standard deviation gives effectively a precise trend for the different tests in the same configuration/surface and can assist in the choice of the optimized modular size with respect to the surface of movement;
- Globally, the configuration with 2 wheels, nevertheless the lowest weight, has the worst behavior with vibrations amplified of 2 times for every surface; even, the movement is not allowed with 2 wheels configuration on very irregular surfaces (like the gravel here considered);
- The configurations with 4 or 6 wheels, have similar behavior for all the tests, only a light preference may be given to the configuration with 6 wheels on soft surfaces (such as the grassland here considered), where probably the use of 2 more wheels may reduce the fluctuations and may improve the LiDAR scanning.

Tab. 2. Standard deviation σ of the vertical acceleration

Environment: Asphalt	Two-wheel type	Four-wheel type	Six-wheel type
Test 1	0.99	0.42	0.37
Test 2	0.97	0.35	0.37
Test 3	1.1	0.37	0.39

Tab. 3. Standard deviation σ of the vertical acceleration

Environment: Grassland	Two-wheel type	Four-wheel type	Six-wheel type
Test 1	0.95	0.42	0.37
Test 2	1.1	0.45	0.35
Test 3	0.93	0.41	0.37

Tab. 4. Standard deviation σ of the vertical acceleration

Environment: Gravel	Two-wheel type	Four-wheel type	Six-wheel type
Test 1	-	0.53	0.48
Test 2	-	0.48	0.49

In order to conclude this analysis about the different configurations also an energetic evaluation of the new robot has been carried out evaluating, from the motor power, the total robot weight and plausible hypothesis of movement, the autonomy of the robot. It has been estimated an autonomy equal to 25 minutes for the configurations with 6 wheels, and 32 minutes for the 4 wheels robot with the actual battery of 3800mAh at 24 Volt. These estimated autonomy times have been confirmed by the experiments conducted.

3.2. The Use of LiDAR Sensor: Preliminary Tests and Pre-Processing of the Data

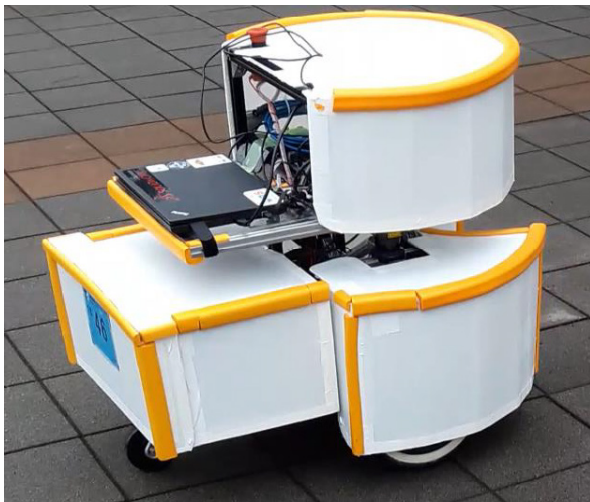
The LiDAR sensor may generate frames containing Point Cloud Data (PCD) mapping the neighboring environment with a sampling frequency of about 5 Hz (5 frames for second). Preliminary tests have been carried out by using the LiDAR data produced by the robot during its exploration. In Fig. 10 an environment placed inside the University Campus (Kyushu Institute of Technology, Tobata Campus), also localized in Fig. 10 from satellite image, is shown. In Fig. 11 b photo of the test setup where is evident that in the environment there is also a second robot (Fig. 11 a) that has the function of tracking object to be detected by the proposed robot, used in the configuration with 4 wheels in order to test the proposed strategy in conditions similar to the real ones. The extracted point clouds related to a frame are shown in Fig. 12a. In Fig. 12b the same frame is visualized after a preliminary pre-processing of the point clouds. The preliminary pre-processing makes the following operations:

- modifying the frame orientation in such a way to align the image to the robot, with respect to the Lidar position on it;
- executing a preliminary filtering of the data eliminating the outliers and the support ground;
- visualization of the resulting pre-processed PCD.

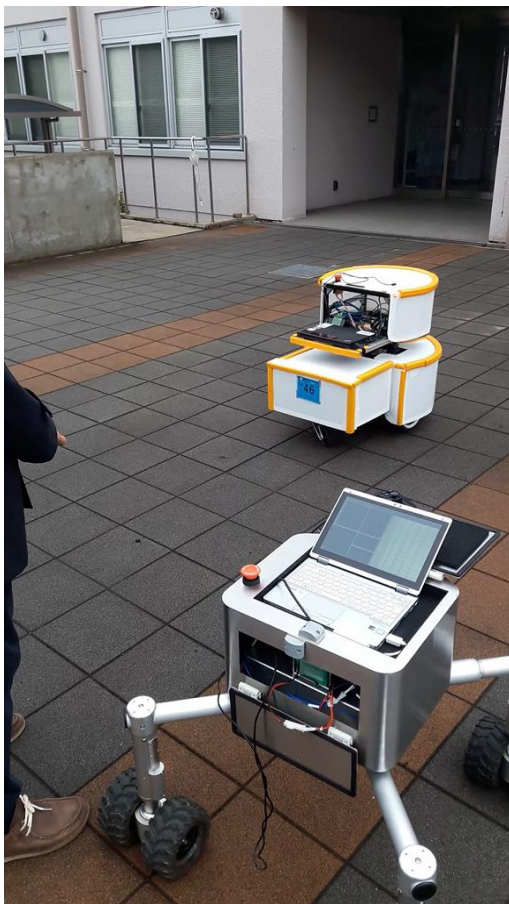
In Fig. 12b it is possible to note the absence of the supporting ground and of outliers. The PCDs after pre-processing could be more useful for giving correct information about the surrounding environment. The pre-processing phase has the aim to eliminate noise sources from the PCDs for facilitating the following phase of features and target search and it will be applied to all the analysis presented in the following section. After the alignment of the LiDAR to the robot (i.e. z axis has to be inverted due to the sensor position), the outliers are removed by considering a maximum distance threshold from the other points. Then, the ground points are eliminated, by considering a threshold of minimum distance by the fitting xy plane that permits to eliminate the points referred to the ground. Finally, the points included in a sphere, having the center in the axis origin (center of the robot) and radius equal to 1 meter, have been eliminated because the LiDAR sensor detects some lateral projections of the sensor included in the visible field of the sensor.



Fig. 10. Environment of preliminary tests

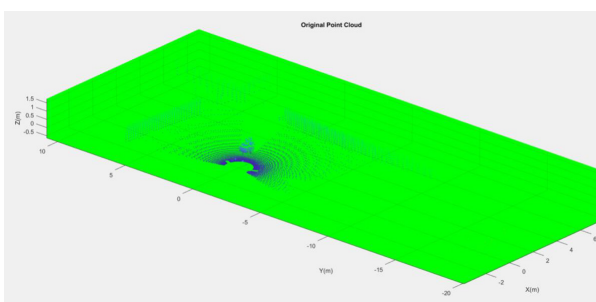


(a)

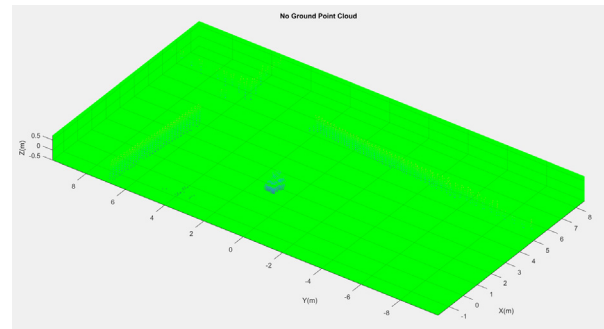


(b)

Fig. 11. a) target robot b) test setup



(a)



(b)

Fig. 12. a) Extracted point clouds, b) Point clouds after pre-processing

4. Algorithm for Robot Odometry and Target Following Based on LiDAR Data

The use of a LiDAR sensor alone or in conjunction with other sensors, may be very useful for determining with accuracy the position of the robot and/or the presence and relative position of external obstacles. The sensor has been installed (see Fig. 5) on the external surface of the robot permitting a simplicity in the signals receptions and in the landmarks chosen.

The basis principle of the algorithm here introduced is similar to that used for solving Simultaneous Localization and Mapping (SLAM) [19-20] problems but considering as landmark of the signals extracted by the LiDAR frames, fixed structures such as walls. The proposed algorithm is divided in two steps:

- the first step operates on each single frame acquired from the LiDAR sensor during the robot exploration giving as result the position and pose of the mobile robot with respect to the researched landmark;
- the second step operates on the sequence of frames related to a specific mapping test and gives as results the trajectory of the robot during the test with respect to landmarks (in fixed positions) and targets (in fixed or also mobile positions).

The algorithm is mainly efficient for tests indoor or where it is easy to individuate some fixed reference structures (walls) that may be used as landmarks for estimating the robot pose. The idea of the proposed algorithm is to identify vertical walls, estimating their intersection and considering this line as the reference for defining the spatial coordinates of the proposed robot. In this way, considering a sequence of n frames recorded by the LiDAR sensor, applying sequentially the algorithm by means of a 'while' cycle, it is possible to estimate the relative coordinates (x,y,z) of the landmark with respect to the sensor position, placed on the robot. The relative coordinates (x,y,z) of the identified landmark are stored in a matrix $n \times 3$ (where n is the number of PCDs saved during the test) and then processed for calculating the robot route. The post-processing proposed forecast the recognition of two vertical walls included in the considered PCD

[21]; the maximum distance between inliers and fitting plane has been fixed at 0.5 meters. After some tests, this chosen has been considered not robust and another condition has been added: the iterative plane fitting process is concluded only if the fitting error (from command 'pcfit' [21]) is lower than a threshold (chosen equal to 0.065 in the considered tests). Moreover, in order to make the algorithm more robust in indoor tests, a further condition of orthogonality between the identified planes is added when the presence of orthogonal walls is guaranteed.

Preliminarily to the walls identification, a partition of the points of the analyzed PCD is necessary; at this purpose a clustering procedure has been considered in the proposed algorithm. The particularity of the clustering procedure here proposed lies in repeating iteratively the procedure for 10 times in such a way to have a robust partition of the data and to consider as starting points for each frame considered the final centroids obtained for the previous frame. This second shrewdness is plausible considering that the target objects (walls and the second robot) have a low velocity of movement with respect to the frame frequency and so, likely, the position of the centroids of each object does not change so much between one frame and the following one (after 0.2 seconds). In this way, the computational efficiency of the clustering procedure increases and gives stables results.

The most important problem in the clustering operation is linked to the necessity to select the number of clusters in which the data has to be classified. All the tests presented in this research have a fixed number of clusters equal to 3, (2 for the walls and one for the robot target) and the clustering is realized by means of the 'kmeans' [19] command. Anyway, an interesting future development of the research is that of finding an auto-adaptive procedure in such a way to automatize the choose of the number of clusters useful for the classification.

In Fig. 13 the fitted plane, Plane 1 from Cluster 1 and Plane 2 from Cluster 2, obtained by applying the described procedure to the frame shown in Fig. 12b. It is possible to note that the clustering has efficiently divided the data in 3 groups; red points (Cluster 1), blue points (Cluster 2) and green points (Cluster 3, target robot) for the point clouds in Fig. 12 b. In Fig. 13 are also shown, with a black cross, the centroids positions for each cluster and with a black sign. Once estimated the two planes from the point clouds, the algorithm calculates the intersection point between these points and the ground fitting plane calculated in the pre-processing phase (paragraph 3.2). The coordinates (x,y,z) of this point are saved and compared with the robot spatial coordinates at the previous step and indicated with a black sign in Fig. 13.

The algorithm has been implemented in such a way to separate the clusters that identify the landmarks (Cluster 1 and Cluster 2) with the cluster related to the target robot (Cluster 3) by checking the distance of the target centroid with respect to the fitting landmark planes.

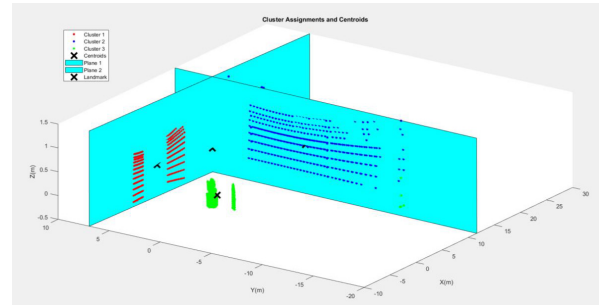


Fig. 13. Algorithm results on a singular frame

The algorithm has been conceived with the aim of solving two tasks:

- Locating, in the surrounding environment, the fix and mobile targets;
- Evaluating the mobile target kinematic in such a way to define the trajectory and foreseeing the future evaluation of the trajectory from the previous frames analysis.

The algorithm has been generalized in such a way that it can process a sequence of consecutive PCD with the described procedure and estimating, for each frame, the fitted plane 1 and 2 and the robot target. . In this way it is possible to estimates the movement of the robot with respect to the fixed reference landmarks that are the walls of the room or of the space where the robot is moving and with respect to the target robot.

4.1. Use of the Proposed Algorithm for Robot Odometry With Stationary Target Robot

One application of the presented algorithm is shown in this test, where the robot in a 4 wheels configuration is moved by a joystick with a forward movement (along x axis). In the reality, the robot has moved of about 5 meters along x-axis, with a small deviation along y-axis and with a null deviation along z axis because the movement were along a flat surface. The application of the proposed procedure give the results depicted in Fig. 15, where the trajectory of the robot is clearly reconstructed (red circles) respecting the experimental behavior, and the target robot position is stationary and corresponds to the points of the figure not included in Plane 1 and Plane 2. The projection of the algorithm results on the xy plane is shown in Fig. 15 where is possible to emphasize the correct reconstruction of the robot trajectory. About the detection of the position of the target robot, the elements of the displacements from the landmark of the robot are equal and opposite to the displacements from the centroid of the target cluster, demonstrating the stationary condition of the target.

Several experimental tests have been carried out, demonstrating, in all the cases, the capacity of the algorithm to estimate the robot trajectory and to detect the stationarity of the target.

It is important to underline that the IMU data collected in all the tests, show important noise, making impossible any trajectory reconstruction close to the reality.

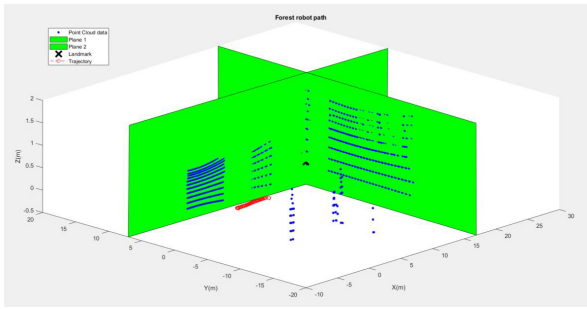


Fig. 14. Algorithm results on consecutive frames

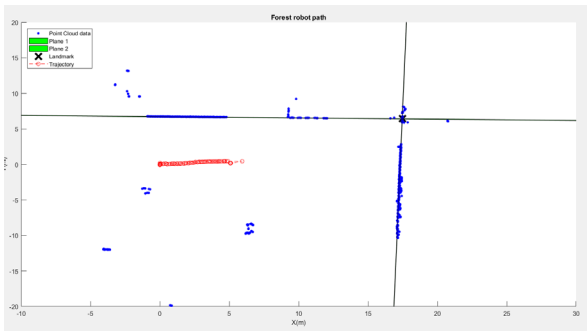


Fig. 15. Projection of the results on xy plane

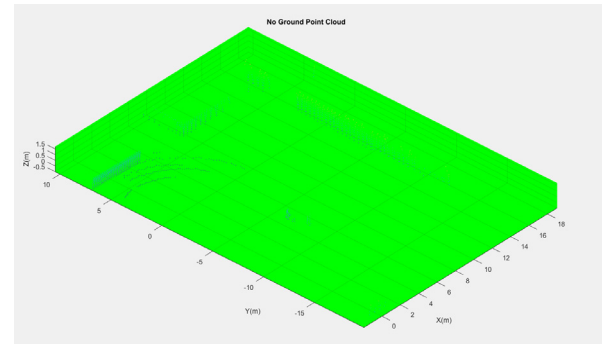
4.2. Use of the Proposed Algorithm for Robot Odometry With Moving Target Robot

In order to test the capacity of the proposed algorithm of tracking mobile targets moving in the visual field of the LiDAR sensor, other tests have been carried out moving the mobile target robot (shown in Fig. 10a) during the movement of the robot equipped with LiDAR.

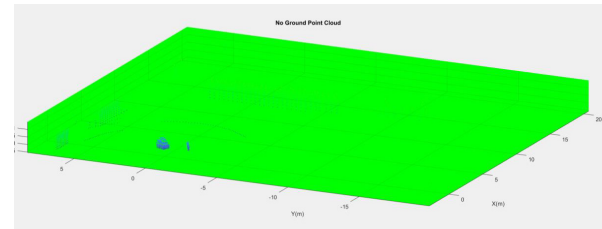
In the following Figures. 16-18 the results of the proposed algorithm referred to one test where the two mobile robots started to move after 10 seconds and stopped after 20 seconds. The first robot (we can name it First robot) moves with a trajectory directed along x axis, the second one, starting from a distance of about 10 meters, is manually moved almost perpendicularly to the direction of First Robot almost crossing its trajectory and stopping very close to it.

The point clouds referred to the starting situation and the final one are reported in Fig. 16, the trajectories reconstructed with the proposed algorithm are depicted in Fig. 17 (red for First robot, blue for the target robot), and the distances component along the reference axis between the two robot, easily calculated after the trajectory reconstruction, are shown in Fig. 18. The proposed algorithm demonstrated its ability to detect mobile elements within the inspection zone and to evaluate the route by using the LiDAR data and the PCDs processing.

The possibility of estimating in real time the trajectories of the target robot can also open interesting scenarios for predicting with a good accuracy its position if its movement continues without brusque accelerations. This possibility is very interesting considering autonomous inspection of autonomous machines equipped with a LiDAR sensor.



(a)



(b)

Fig. 16. a) Point clouds related to the starting position of the test, b) Point clouds related to the final position of the test

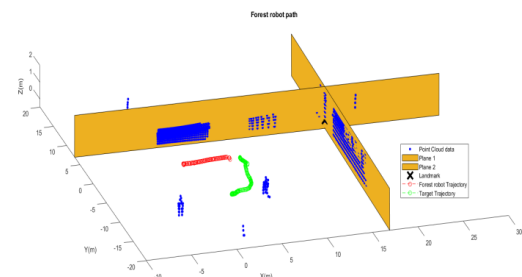


Fig. 17. Algorithm results showing the calculated trajectories with mobile target robot

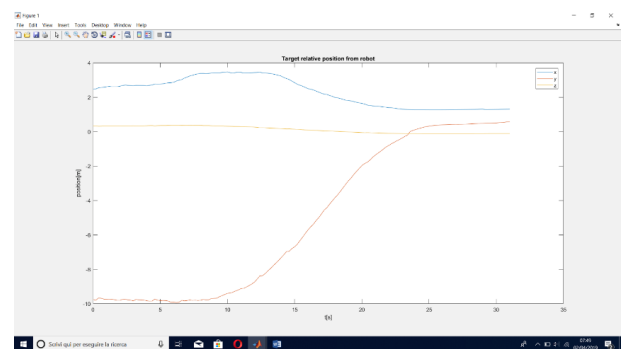


Fig. 18. Components of the distances between the robots in the test with mobile target robot

5. Conclusion

In this paper, a modularized driving wheels robot able to have a balance between the mobile robot's runnability and efficiency according to the environ-

ment has been presented together with an efficient algorithm using LiDAR data for robot self-localization and detection of mobile targets. The developed robot, which can be constructed by combining modules, may travel through several environment with saving the energy by adaptively change the number and arrangement of the wheels according to the environment. Moreover, several experiments have been conducted for evaluating the performance and the characteristics of the developed robot. First, it was shown that the developed mobile robot could easily change the three types of mechanisms according to the environment by changing the number of modularized driving wheels and their combination. The experimental tests confirmed that the six-wheel type is suitable for an uneven and soft environment and the configuration with 4 wheels has similar characteristics and bigger autonomy for hard and rigid surfaces. Next, it was shown by testing this mobile robot that it is possible to use the LiDAR data for constructing a 3D map by running on structured environment. In addition, a specific algorithm for automatically analyzing the LiDAR data has been presented and tested, demonstrating that, in specific structures scenarios, the robot can self-localize its position compared to fixed landmarks and may also evaluate the movement of eventual mobile targets, foreseeing their next displacements.

Acknowledgements

The authors thank the students Luigi Ricciato, Alesio Monaco and Kakeru Yamashita for their precious work related to this research.

AUTHORS

Nicola Ivan Giannoccaro* – Department of Innovation Engineering, University of Salento, Lecce, 73100, Italy, e-mail: ivan.giannoccaro@unisalento.it.

Takeshi Nishida – Department of Control Engineering, Kyushu Institute of Technology, Kitakyushu, Japan, e-mail: nishida@cntl.kyutech.ac.jp.

*Corresponding author

REFERENCES

- [1] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator". In: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, 2004, 2149–2154, DOI: 10.1109/IROS.2004.1389727.
- [2] P. Forsman and A. Halme, "3-D mapping of natural environments with trees by means of mobile perception", *IEEE Transactions on Robotics*, vol. 21, no. 3, 2005, 482–490, DOI: 10.1109/TRO.2004.838003.
- [3] T. Tsubouchi, A. Asano, T. Mochizuki, S. Kondou, K. Shiozawa, M. Matsumoto, S. Tomimura, S. Nakanishi, A. Mochizuki, Y. Chiba, K. Sasaki and T. Hayami, "Forest 3D Mapping and Tree Sizes Measurement for Forest Management Based on Sensing Technology for Mobile Robots". In: K. Yoshida and S. Tadokoro (eds.), *Field and Service Robotics: Results of the 8th International Conference*, 2014, 357–368, DOI: 10.1007/978-3-642-40686-7_24.
- [4] J. Billingsley, A. Visala, M. Dunn, "Robotics in Agriculture and Forestry". In: B. Siciliano and O. Khatib (eds.), *Springer Handbook of Robotics*, 2008, DOI: 10.1007/978-3-540-30301-5_47.
- [5] X. Liang, P. Litkey, J. Hyypä, H. Kaartinen, M. Vastaranta and M. Holopainen, "Automatic Stem Mapping Using Single-Scan Terrestrial Laser Scanning", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 2, 2012, 661–670, DOI: 10.1109/TGRS.2011.2161613.
- [6] M. A. Juman, Y. W. Wong, R. K. Rajkumar and L. J. Goh, "A novel tree trunk detection method for oil-palm plantation navigation", *Computers and Electronics in Agriculture*, vol. 128, 2016, 172–180, DOI: 10.1016/j.compag.2016.09.002.
- [7] R. A. Chisholm, J. Cui, S. K. Y. Lum and B. M. Chen, "UAV LiDAR for below-canopy forest surveys", *Journal of Unmanned Vehicle Systems*, 2013, DOI: 10.1139/juvs-2013-0017.
- [8] M. Morita, T. Nishida, Y. Arita, M. Shige-eda, E. di Maria, R. Gallone and N. I. Giannoccaro, "Development of Robot for 3D Measurement of Forest Environment", *Journal of Robotics and Mechatronics*, vol. 30, no. 1, 2018, 145–154, DOI: 10.20965/jrm.2018.p0145.
- [9] K. Kamikawa, T. Arai, K. Inoue and Y. Mae, "Omni-directional gait of multi-legged rescue robot". In: *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 3, 2004, 2171–2176, DOI: 10.1109/ROBOT.2004.1307384.
- [10] S.-A. Li, H.-M. Feng, K.-H. Chen, J.-M. Lin and L.-H. Chou, "Auto-maps-generation through Self-path-generation in ROS-based Robot Navigation", *Journal of Applied Science and Engineering*, vol. 21, no. 3, 2018, 351–360, DOI: 10.6180/jase.201809_21(3).0006.
- [11] M. G. Ocando, N. Certad, S. Alvarado and Á. Terrones, "Autonomous 2D SLAM and 3D mapping of an environment using a single 2D LIDAR and ROS". In: *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*, 2017, DOI: 10.1109/SBR-LARS-R.2017.8215333.
- [12] Y.-T. Wang, C.-C. Peng, A. A. Ravankar and A. Ravankar, "A Single LiDAR-Based Feature Fusion Indoor Localization Algorithm", *Sensors*, vol. 18, no. 4, 2018, DOI: 10.3390/s18041294.

- [13] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler and A. Ng, "ROS: an open-source Robot Operating System", *Proceedings of ICRA Workshop on Open Source Software*, 2009.
- [14] D. Shrivastava, "Designing of All Terrain Vehicle (ATV)", *International Journal of Scientific and Research Publications*, vol. 4, no. 12, 2014.
- [15] D. Pradhan, J. Sen and N. B. Hui, "Design and development of an automated all-terrain wheeled robot", *Advances in robotics research*, 2014, 21–39, DOI: 10.12989/arr.2014.1.1.021.
- [16] R. Shah, S. Ozcelik and R. Chaloo, "Design of a Highly Maneuverable Mobile Robot", *Procedia Computer Science*, vol. 12, 2012, 170–175, DOI: 10.1016/j.procs.2012.09.049.
- [17] D. B. Bickler, US Patent Number 4,840,394, "Articulated Suspension Systems", US Patent Office, Washington, DC, 1989.
- [18] B. D. Harrington and C. Voorhees, "The Challenges of Designing the Rocker-Bogie Suspension for the Mars Exploration Rover". In: *Proceedings of the 37th Aerospace Mechanisms Symposium*, 2004.
- [19] S. Thrun, W. Burgard and D. Fox, *Probabilistic robotics*, MIT Press, 2005.
- [20] J. Tang, Y. Chen, A. Kukko, H. Kaartinen, A. Jaakkola, E. Khoramshahi, T. Hakala, J. Hyyppä, M. Holopainen and H. Hyyppä, "SLAM-Aided Stem Mapping for Forest Inventory with Small-Footprint Mobile LiDAR", *Forests*, vol. 6, no. 12, 2015, 4588–4606, DOI: 10.3390/f6124390.
- [21] "ComputerVisionToolbox". Mathworks, <https://www.mathworks.com/products/computer-vision.html>. Accessed on: 2021-02-03.