VOLUME 16, N° 1 2022
Journal of Automation, Mobile Robotics and Intelligent Systems

Processing of LiDAR and IMU data for target detection and odometry of a mobile robot

Submitted: 10th June 2021; accepted: 2nd August 2022

Nicola Ivan Giannoccaro, Takeshi Nishida, Aimè Lay-Ekuakille, Ramiro Velazquez, Paolo Visconti

DOI: 10.14313/JAMRIS/1-2022/1

Abstract:
In this paper, the processing of the data of a 3D light detection and distance measurement (LiDAR) sensor mounted on a mobile robot is demonstrated, introducing an innovative methodology to manage the data and extract useful information. The LiDAR sensor is placed on a mobile robot which has a modular design that permits the easy change of the number of wheels, was designed to travel through several environments, and saves energy by changing the number and arrangement of the wheels in each environment. In addition, the robot can recognize landmarks in a structured environment by using a classification technique on each frame acquired by the LiDAR. Furthermore, considering the experimental tests, a new simple algorithm based on the LiDAR data processing together with the inertial data (IMU sensor) through a Kalman filter is proposed to characterize the robot's pose by the surrounding environment with fixed landmarks. Finally, the limits of the proposed algorithm have been analyzed, highlighting new improvements in the future prospective development for permitting autonomous navigation and environment perception with a simple, modular, and low-cost device.

Keywords: *mechatronics device, LiDAR sensor, measurement processing, 3D object recognition, robotic implementation.*

1. Introduction

LiDAR sensors have been firmly established as an essential component of many mapping, navigation, and localization applications for autonomous vehicles, as demonstrated by the increasing number of recent studies focused on the usage of LiDAR data for different applications. This popularity is mainly due to the improvements in LiDAR performance in terms of range detection, accuracy, power consumption, and physical features such as dimension and weight. This technology is considered to be state of the art, considering that only very recent contributions permit us to compare different algorithms, models, methodologies, and techniques for different applications of LiDAR in the autonomous vehicles. Goelles et al. present a systematic review on faults and suitable detection and recovery methods for automotive perception sensors with a focus on LiDAR to review the state-of-the-art technology and identify promising research opportunities [1]. Raj et al. explore LiDAR scanning mechanisms employed in LiDAR technology from past research to current commercial products, revealing that electro-mechanical scanning is the most prominent technology in use today [2]. Another recent review provides a comprehensive survey of the Simultaneous Localization and Mapping (SLAM) used for many applications, including mobile robotics, self-driving cars, unmanned aerial vehicles, or autonomous underwater vehicles, focusing on solutions a sensor fusion between vision and LiDAR [3].

As demonstrated by recent reviews, applications of LiDAR are becoming significant, and it is important to acquire new ideas and algorithms related to the use of this data. Among the various fields of LiDAR application, mobile robots form an important category. Mobile robots often have to analyze the surrounding environment to get information about the presence of objects and the trajectory of the robot concerning landmarks or mobile targets. Often in applications, it is necessary to measure the environment in detail when it is difficult to identify the position by global navigation satellite system (GNSS) signals; in these cases, the generation of the three-dimensional digital map by LiDAR is a good solution, as shown in several papers [4-11]. Methods of measuring the surrounding environment using 2D LiDAR have been proposed for four-wheeled mobile robots [7-8], for an unmanned aerial vehicle (UAV) [7], and for mobile robots with a rocker-bogie mechanism [8]. Various techniques of processing and filtering the sensor's data have also been studied and presented [12].

This research focuses on a simple, low-cost robot designed and introduced by Giannoccaro et al. in 2019, driven by ROS and realized with a modularized wheel and a frame for connecting them and for supporting a scanning LiDAR [13]. A Lidar Hokuyo 3D sensor, which has been increasingly used in recent years, especially for mapping and operations in the automation field, is installed on the robot case. Moreover, an inertial measurement unit (IMU sensor) is present on the robot, and supplies the three linear accelerations along the three axes [x, y, z] and the three angular accelerations along the same axes with a given sampling frequency.

The objectives of the research work are to provide an algorithm capable of estimating the odometry of the mobile robot with high reliability during the journeys made. This practice is very important in the robotic field: in the literature, there are various SLAM algorithms and, more generally, algorithms aimed at reconstructing the vehicle's trajectory. In most cases, IMU sensors are used alongside either GPS data or, more rarely, data provided by fixed cameras that inspect a certain control environment to obtain the input data of the algorithms. It is therefore clear that both methodologies have limits. In the first case, the method would not apply to indoor environments to be inspected due to the difficulty of the GPS signal to reach the robot, while in the second case, the area to be inspected should be equipped with element support, such as cameras or other types of sensors, binding the research to very limited environments. Preliminary results obtained by using only LiDAR data for the proposed robot are shown in a 2020 study by Giannoccaro et al. [14]. The algorithm presented here, on the other hand, uses the IMU sensor and the LiDAR sensor, with the latter being capable of providing the frames containing point cloud data with a defined sampling rate, thus characterizing the neighboring environment. This makes it more versatile in eliminating critical issues, with a similar methodology to that used for an array of ultrasonic sensors for scanning external environments, determining useful characteristics for the reconstruction of simple external reference surfaces [15-17]. The algorithm makes use of a linear Kalman filter, which evaluates the dynamic state of a system starting from measurements affected by noise. In the test proposed, in the Kalman filter, greater weight was given to the filter correction phase using the LiDAR observations compared to the prediction phase, because the IMU model installed on the robot was not suitable for estimating the installation due to its noise level, as demonstrated by the first preliminary experimental tests. With an opportune preliminary sensor accuracy evaluation, the statistical proposed algorithm could be adapted and generalized to every situation.

The use of the Kalman filter for improving the fusion of the data from LiDAR and other inertial sensors has recently become a highly used and investigated modality for locating and estimating the position of robots or mobile vehicles. Some contributions recently published show innovative procedures that use this strategy for specific applications. In one paper, the author proposes a method for autonomous cars localization based on the fusion of LiDAR and radar measurement data using the unscented Kalman filter for object detection; in this case, the LiDAR and rad fusion obtained by the Kalman filter provide pole-like static object pose estimations that are well suited to serve as landmarks for vehicle localization in urban environments [18]. In another paper, the authors introduce an odometry procedure based on the fusion of LiDAR feature points with IMU data using a tightly coupled, iterated, extended Kalman filter to allow robust navigation in UAV flight [19].

In a paper by Chauchat et al., the authors propose a localization algorithm based upon the Kalman filter to improve smoothing on the fusion of IMU and LiDAR data successfully tested on an equipped car [20]. Muro et al. present a method for moving-object detection and tracking using a LiDAR mounted on a motorcycle and the distortion in the scanning LiDAR data is corrected by estimating the pose of the motorcycle in a period that the LiDAR scan period using information from an IMU via the extended Kalman filter [21]. Zhang et al. introduce a multi-level sensor data fusion method for real-time positioning and mapping [22]. In this method, coordinates are transformed based on the pre-integration results of IMU data in data pre-processing to register the laser point cloud. Features of the laser point cloud are sampled to reduce the computation of point cloud matching. Next, the robot pose is obtained by combining IMU and LiDAR observations with an unscented Kalman filter algorithm to improve the loop closure detection effect.

In another paper, the authors propose an adaptive pose fusion (APF) method to fuse the robot's pose and use the optimized pose to construct an indoor map [23]. Firstly, the proposed method calculates the robot's pose by the camera and inertial measurement unit (IMU), respectively. Then, the pose fusion method is adaptively selected according to the motion state of the robot. When the robot is in a static state, the proposed method directly uses the extended Kalman filter (EKF) method to fuse camera and IMU data, and when the robot is in a motive state, a weighted pose fusion method is used to fuse camera and IMU data. Next, the fusion-optimized pose is used to correct the distance and azimuth angle of the laser points obtained by LiDAR.

Morita et al. propose a robust model predictive control for the trajectory tracking of a small-scale autonomous bulldozer in the presence of perturbations; the pose estimation, for control feedback, is based on sensor data fusion performed by an extended Kalman filter, which processes inertial measurement unit (IMU) and light detection and ranging (LiDAR) measurements [24]. In another paper, the authors introduce a methodology for positioning and autonomous navigation of Unmanned Surface Vehicle based on two-dimensional lidar combined with IMU to perceive the surrounding environment, and the Extended Kalman Filter algorithm is used for the fusion of map matching data and IMU pre-integration data [25].

In this work, the use of the Kalman filter for combining IMU and LiDAR data is applied to a mobile robot specifically built with the characteristics of low cost, modularity, and simplicity, with the aim of using the detection of targets (fixed vertical flat planes), often available in real scenarios. For an indoor scenario, this could be the walls of rooms, or, for an outdoor scenario, the external walls of a building, a deposit, a house of tools, etc., for improving the estimation of the robot trajectory. The proposed procedure is simple, without relevant post-processing of the IMU and LiDAR data, and it uses the vertical planes to simplify the post-processing that requires only classical tools and a low computational effort also compatible with a low-cost solution such as the proposed one.

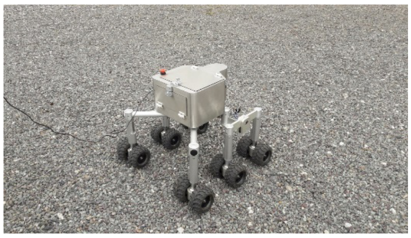
This paper will deal with the following topics: in Section 2, a description of the mobile robot and the considered LiDAR and IMU sensors; in Section 3, a strategy for a pre-processing of the LiDAR data; in Section 4, the innovative algorithm based on LiDAR and IMU data for odometry and target recognition; in Section 5, the results of experimental tests; and, finally, in Section 6, the conclusions of the work.

2. Description of the mobile robot and the analyzed LiDAR and IMU sensors

The mobile robot prototype used was entirely designed and built in the laboratories of the Kyushu Institute of Technology [13, 14]. It is shown in Figures 1 and 2, and it has the specific characteristic that its frame has been designed so that the number of drive wheels can be changed according to the environment in which the robot is travelling. Depending on the environment, the robot can be easily changed by the user to two energy-efficient wheel versions, six-wheel versions with high running ability against complicated terrain, or four-wheel versions with intermediate capabilities [13, 14]. The aim of this modularized, low-cost asset was to minimize the oscillations on the robot and minimize the errors of acquisition of sensors mounted on the robot during exploration in unstructured environments such as a forest environment [24]. The structure has an almost cubic, hollow case, which houses part of the electronic components for the user to control the robot, as well as the battery pack and a red safety button. In addition to the internal cavity, there is also a cavity on the upper part accessible by opening a hatch. This inlet is used mostly to lay the terminal with which to launch the various codes for the management of locomotion and data recording. The LiDAR sensor with which the robot is equipped is the Hokuyo XVT-35 LX 3D LiDAR sensor model [25]; it is installed in the front part of the case (Fig. 2) and housed in a structure designed to avoid any knocks and damage to the sensor during the robot's motion [13, 14]. The solid angle that can be inspected with this LiDAR model is provided by a maximum horizontal angle of 210° and a maximum vertical angle of 35° (the central vertical axis is 15° over the horizon). Regarding the inertial measurement unit, the model MPU-6500 is installed on the robot [26]. The IMU sensor is fixed to the robot by bolting to avoid unwanted movements by the unit that would lead to incorrect measurements, and is located in the front of the case (Figure 2), near the LiDAR sensor and in an area as free from vibrations as possible.

The locomotion of the robot is due to two main circuits: a power supply circuit and a drive wheel control circuit. For the robot, each of the driving wheels is independently powered by a D.C. motor. Each D.C. motor is powered by a Cytron MDD10A board [27]. The MDD10A board may control two high-current D.C. motors, supplying up to 10A continuously. As for the control circuit, the Arduino MEGA electronic board is used. Each power supply board is capable of controlling two motors; therefore, a total of six PWM outputs are required for speed control and six digital outputs are required for direction control. The Arduino MEGA electronic board is used for the control circuit. A scheme of connection for the controller and motors is shown in Figure 3; the positive board terminal is connected to an emergency stop button referred to at the beginning of this chapter, also known as the "mushroom" button. During operation, the button maintains the continuity of the positive power cable, but, if pressed, immediately interrupts it, opening the power circuit so that the robot stops. Finally, the battery fitted to the prototype is a 24V 3800mAh Batteryspace Panasonic.

a)



b)

Fig. 1. a) Six wheels configuration b) Four wheels configuration

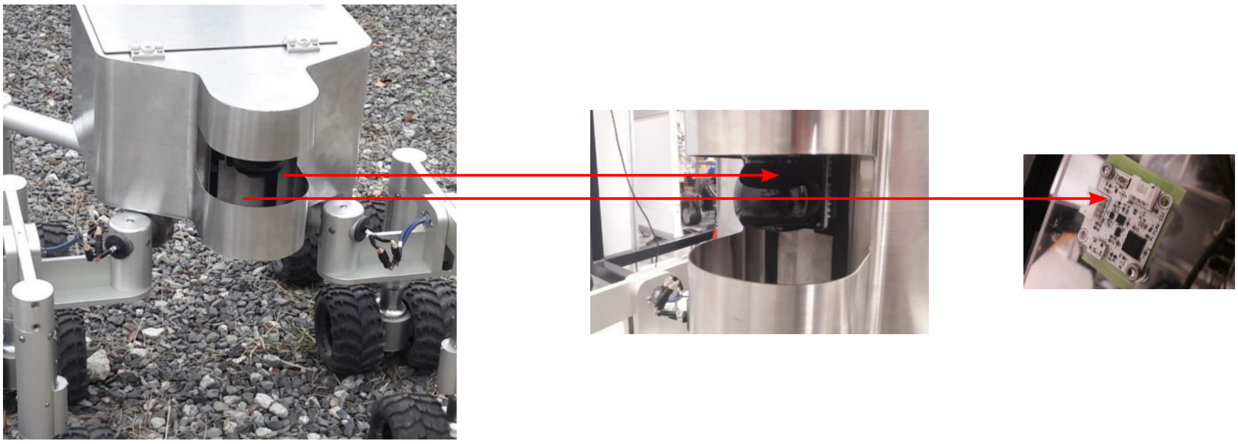


Fig. 2. Front of the robot and protective structure for placing the LiDAR sensor and IMU sensor

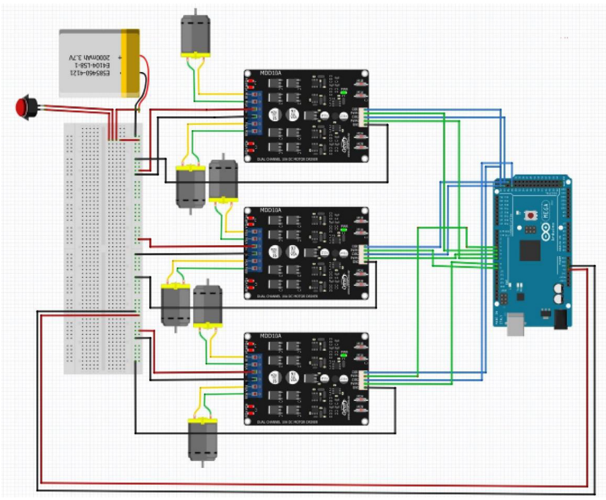


Fig. 3. Scheme of connection for the locomotion and control system

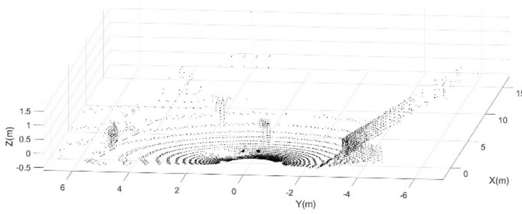
To control the robot, one laptop computer with Intel Core m5 (1.10 GHz) and Ubuntu 14.04 LTS has been used, placing it at the top of the robot (as shown in Fig. 1 a) and connecting it to the boards and sensors through a USB cable. The robot operating system (ROS) [28] was adopted as middleware to manage the locomotion and control system and to acquire the data from the LiDAR and IMU sensors. The formats of the files containing IMU data and LiDAR data are different. For the former, a file in ".csv" format shows the linear acceleration data relating to the three axes in the first three columns and the angular velocity data relating to the same axes in the last three. For the latter, at first, a file with the ".bag" extension, which must be subsequently converted into as many ".pcd" files as the number of frames acquired. Each acquisition file includes a reference about the P.C. processor clock to synchronize the different files of the sensors. Finally, a joystick connected to the laptop through the USB gate has been considered to manually driving the robot during the experimental tests below shown.

3. Processing of the LiDAR data

The visualization of the LiDAR data scans may be carried out with various methodologies; one of the most common makes use of the A.R. Viz Virtual Studio software, with which it is possible to dynamically reproduce the captured frames over time [29]. This method has an immediate visual impact but is not functional for the purposes of this research work. Rather, it is necessary to thoroughly process the single frame and then, at a later stage, draw conclusions on the entire duration of the test. To do this, a first code was written in a Matlab environment [30] that performs the rotation of the image 180° concerning the x-axis of the robot (this operation is necessary as the LiDAR is mounted upside down due to the confirmation of the structure in which it is housed, see Figure 2) and the plotting of the resulting Point Cloud Data. Moreover, the first preliminary tests were carried out by moving the robot using a joystick and acquiring it from the Lidar. Some frames of the path in Figure 4 (which coincides with the x-axis of the chosen reference system, with the z-axis upward, and the y-axis completes the cartesian reference system) have been preliminarily analyzed, and one of them is shown in Fig. 5a. The image of a frame (PCD) shows many disturbances in the surrounding environment, including the ground reflection. To this end, a pre-processing denoising and filtering code has been used to eliminate the presence of the supporting ground and to reduce the noise and the disturbing elements in the following target recognition phase. This operation is also of fundamental importance, as the data initially have different outliers that could compromise subsequent processing operations. The effect of pre-processing is evident in Fig. 5 a) and b) where the same frame is shown before and after the pre-processing process: the underlying ground and many disturbances of the surrounding environment disappear, leaving only the tree and the boulders to the left and right sides of the robot.

Fig. 4. Robot path and acquired environment for pre-processing code testing

a)



b)

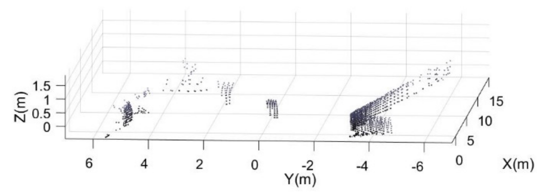


Fig. 5. a) PCD before pre-processing filtering b) PCD after pre-processing filtering

4. Algorithm for robot odometry and target following based on LiDAR and IMU data

The algorithm proposed here can be categorized under the category of Simultaneous Localization and Mapping (SLAM) algorithms, but with some substantial differences: SLAM algorithms aim to estimate the pose of the robot and to carry out the mapping of the environment starting from the checks and observations performed by the robot during exploration. Both types of input data are affected by statistical noise, and this makes it necessary to use statistical filters for the processing of probability densities. The algorithm proposed here considers the use of a linear Kalman filter for updating the iterative process for estimating the pose and from the map. The novelty proposed in this research work consists of the use of a LiDAR sensor to perform the "observations" of the surrounding environment; the most common odometric techniques make use of Dead Reckoning methodologies in which the inertial data are corrected, using a Kalman filter, using data acquired by GPS systems. Another technique is linked to the SLAM algorithms implemented, considering "controls" of an inertial nature and "observations" of the environment performed with laser sensors or fixed cameras that provide the position of the landmarks selected as a reference. It is clear that both solutions have limitations; the use of a GPS is versatile and allows the scanning of very large areas without apparent significant problems, and this sensor is ideal for testing outdoor but could present several complications for indoor explorations due to the difficulty of receiving the signal from the satellite. On the other hand, even the use of laser sensors or fixed cameras for the recognition of features present in the environment is binding, as it would be necessary to equip the environment with certain equipment prior to the exploration phase, which could be difficult to implement in some cases. Furthermore, the inability of these devices to follow the robot's motion limits the exploration area to the places where these elements are present.

The use of a LiDAR sensor to operate the robot observations partially solves the problems just outlined. The sensor is installed directly on the external structure of the robot, and this allows for greater flexibility in the choice of landmarks that may not even be selected upstream of the exploration. Furthermore, there is no difficulty related to the reception of external signals, and the post-processing operations can also be carried out offline for all applications that do not require the autonomous motion of the robot.

4.1. The kinematic model for robot controls using inertial measurements

The controls on the installation of the robot can be carried out using inertial data acquired during the explorations by the IMU sensor that measures the triaxial linear and angular accelerations. A double integration operation must then be performed over time to obtain the three spatial coordinates that will identify the pose of the robot from the measured accelerations. The system and output equations written for a generic dynamic system in the state space form (1) can be particularized for the kinematic of the robot as expressed in (2) and (3), where *x,y,z, vx,vy,vz, ax, ay, az*, are the displacement, velocity and acceleration components, along with the *xyz* cartesian reference system.

$$\begin{cases} \dot{x}(t) = A\,x(t) + B\,u(t) \\ y(t) = C\,x(t) + D\,u(t) \end{cases} \tag{1}$$

$$\begin{bmatrix} v_x \\ v_y \\ v_z \\ a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ v_x \\ v_y \\ v_z \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \tag{2}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ v_x \\ v_y \\ v_z \end{bmatrix} \tag{3}$$

The model has then been discretized by using the forward Euler method, which is among the simplest, and provides a coherent model for sampling sufficiently high frequencies. Naming the sampling period Ts, the discretised kinematic model is expressed in (4) where [I+ATs] and B.T.s are expressed in (5) and (6), taking into account (2), (3).

$$x(k+1)=[I+A\,T_s]x(k)+BT_s\,u(k) \tag{4}$$

$$[I + A\,T_s] = \begin{bmatrix} 1 & 0 & 0 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 & T_s & 0 \\ 0 & 0 & 1 & 0 & 0 & T_s \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

$$[B\,T_s] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ T_s & 0 & 0 \\ 0 & T_s & 0 \\ 0 & 0 & T_s \end{bmatrix} \tag{6}$$

In this way, the matrices to implement the prediction phase of the Kalman filter through the realization of a kinematic model have been obtained.

4.2. The algorithm model for robot pose evaluation using LiDAR data

The basic principle of the proposed algorithm is similar to that of robot observations in SLAM problems, but instead of evaluating the presence and position of landmarks through laser transducers, they are evaluated by applying a specific algorithm to each frame acquired during the exploration by the LiDAR sensor.

The algorithm is mainly efficient for indoor tests, in which it is possible to identify some fixed reference structures concerning where to identify the installation of the robot. In particular, it is necessary to be in the presence of vertical flat walls, as is the case for any indoor test, to evaluate the intersection between two of them and consider this as a landmark to identify the spatial coordinates of the robot's center of gravity. The results that will be shown below refer to outdoor tests, although, as previously stated, the algorithm proposed for robot observations is more efficient for indoor environments, given the almost certain presence of vertical walls that can be taken as a reference. However, an environment of this type is preferred to have more space available for exploration while still having references available to use. The setting chosen for the explorations is one of the courtyards present in the Kyushu Institute of Technology depicted in Fig. 6a; Fig.6b shows an image from above of the courtyard explored. The two vertical walls are highlighted in red in Fig 6b; these walls, in most of the acquired experimental tests, have been taken as a reference for the odometric information of the robot acquired by LiDAR.

a) b)

Fig. 6. a) Location of the tests b) Satellite images of the inspected environment

It is possible to trace the coordinates of the landmark relative to the robot, since for each pcd frame, the origin of the Cartesian axes coincides with the center of gravity of the LiDAR, which is installed on the robot itself. For this reason, it is legitimate to consider, omitting a negligible error, the origin of the axes centered in the center of gravity of the rover. For each LiDAR-acquired frame, the algorithm before performs the pre-processing filtering described in Section 3, then also removes the points present in a sphere with the center at the origin of the axes and a radius of 1 meter. This operation is necessary since the LiDAR also "observes" some lateral protrusions of the robot that fall within the sensor's field of view. The post-processing involves the recognition of two flat vertical walls possibly present in the pcd frame under examination. The function used is also, in this case, the "pcfitplane", providing in input a maximum distance between inliers and fitting plane equal to 0.5 meters [30]. Since, after some tests, it was noticed how this algorithm was not robust and could not capture the plans of interest, a more restrictive condition was inserted: the plan fitting process is stopped only if the mean square error of the inliers points of the model is less than a certain threshold. After this, the algorithm became more robust. In some cases, it is possible to insert a further condition that provides for the orthogonality between the two flat walls (the typical situation for indoor environments) to further strengthen the algorithm. To divide the points, a clustering technique aimed at homogeneous grouping objects starting from a set of data has been used. Clustering techniques are based on measures relating to the similarity between the elements. The discriminating variable for the grouping of the elements depends on the problem proposed; in the case of point cloud data, it is the distance between the points. In this work, the clustering has been carried out by using the "kmeans" function [15-17, 30]. The "kmeans" function takes as input the coordinates of the point cloud on which clustering will be performed and the desired number of partitions, while it outputs the indices of the points indicative of the assignment of points to one of the partitions, the coordinates of the centroids, and the distance of all points from each centroid. In this case, fixing the number of partitions at 3, it is possible to identify the two flat models and the robot feature, and it is possible to calculate the intersection point between these two floors and the fitting ground plane calculated in the pre-processing phase. An example of a frame processing procedure obtained while the robot moved in an indoor environment is shown in Figure 7a and 7b (first and after the preliminary processing), and the consequent walls and landmark detection is shown in Figures 8 and 9. In Figure 8 and, for more clearness Figure 9, the results of the proposed algorithm for one frame acquired by the LiDAR sensor mounted on the robot: the 3 clusters identified through the clustering technique used are indicated with Cluster 1,2 and 3 and their points have been differentiated with different symbols in Fig.7. The centroids of the 3 clusters are indicated with a square; by checking the alignment of the individuated clusters, the walls points are identified and the planes 1 and 2 are estimated by fitting the points. Finally, the landmark is evaluated by considering the intersection of the estimated planes at the level z=0.

a)

b)

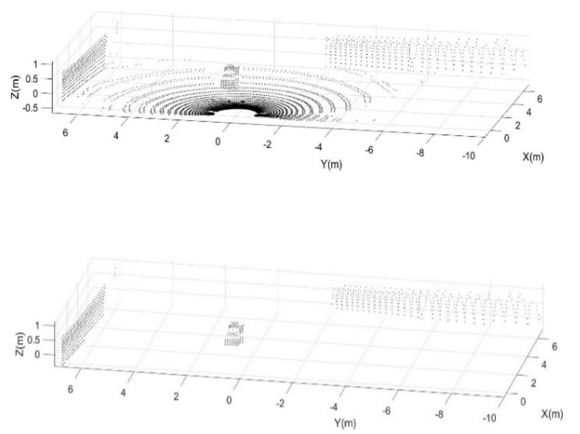


Fig. 7. a) PCD indoor frame before pre-processing filtering b) PCD after pre-processing filtering

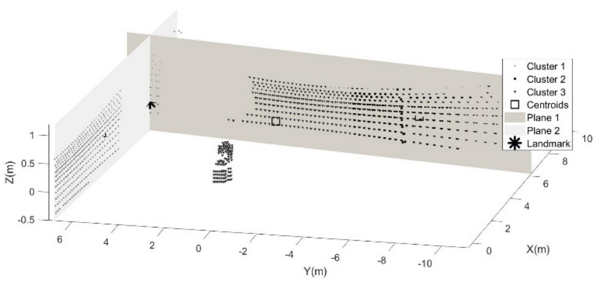


Fig. 8. Identification of the landmark as intersection of the identified walls (planes 1 and 2)

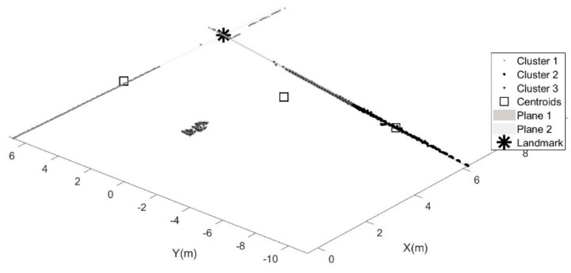


Fig. 9. Top view of the centroids of the 3 identified clusters and landmark estimated for robot position

This procedure may be generalised considering a trajectory traveled by the robot during which the succession of frames relating to different positions are acquired and processed with the technique shown. The detection of the reference point allows the reconstruction of the position of the robot concerning the two walls identified in the sequence of frames during the movement. Taking into account the spatial coordinates of the robot in the previous step, it is possible to calculate the displacement considering the fixed reference in the various subsequent frames. It is therefore possible to structure a while loop that builds the position matrix containing the displacements, in meters, of the robot, starting from the initial position at step k = 1, at coordinates [0,0,0]. The output of the algorithm, therefore, consists of a matrix n x 3, where n is the number of analyzed frames, which expresses the odometry of the robot concerning the identified landmark, the intersection of the identified walls. However, this work aims to propose a methodology for integrating the information on the position of the robot obtained from the LiDAR, with those obtained from the inertial data (IMU) from the sensor mounted on the robot.

4.3. Integration of LiDAR and IMU data through a discrete Kalman filter

Considering the possibility of having odometric data simultaneously from LiDAR and IMU, it was decided to integrate the information using a discrete Kalman filter, in which the information of one of the sensors is considered an estimate of the position of the robot, while the information of the other constitutes the updated feedback. The prediction equation for the kinematic model expressed in (4)-(6) is shown in (7), where P(k) is the error covariance matrix at step k, and Q is the process noise covariance matrix. The correction equations are shown in (8), where R is the measurement noise covariance matrix, and C derives from the state model (1).

$$\begin{cases} \hat{x}(k)^- = A\,\hat{x}(k-1) + B\,u(k) \\ P(k)^- = A\,P(k-1)A^T + Q \end{cases} \tag{7}$$

$$\begin{cases} L(k) = P(k)^{-} C^{T} \left[C\, P(k)^{-}\, C^{T} + R\right]^{-1} \\ \hat{x}(k) = \hat{x}(k)^{-} + L(k)\left[y(k) - C\, \hat{x}(k)^{-}\right] \\ P(k) = \left[I - L(k)C\right] P(k)^{-} \end{cases} \tag{8}$$

For the evaluation of the system noise related to the inertial sensor and more generally for the calibration of the sensors, the results of various tests performed with the robot completely stopped are evaluated. From the trends of the measured accelerations, it can be seen that all channels are affected by ample noise. In particular, the channel relating to the y axis is the one with a higher standard deviation of the data and, therefore, the loudest. The covariance matrices of process noise Q and measurement noise R used for the tests shown in the next paragraph are reported in (9) and (10) and take into account the characteristics of the two sensors' calibration data.

$$Q = \begin{bmatrix} 0.1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.1 \end{bmatrix} \tag{9}$$

$$R = \begin{bmatrix} 10^{-3} & 0 & 0 \\ 0 & 10^{-3} & 0 \\ 0 & 0 & 10^{-3} \end{bmatrix} \tag{10}$$

Given the high noise encountered during the calibration phase in the inertial measurement unit (IMU), by providing these covariance matrices as input, it was decided to give greater weight to the corrections made with LiDAR measurements. The matrices must therefore be adapted and chosen based on the characteristics of the sensors available. The output provided by the Kalman filter consists of the laying of the robot as a function of time starting from the inertial controls of the robot, corrected by the observations made by the LiDAR with a frequency of about 5 Hz.

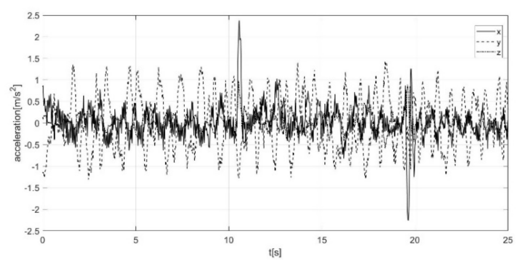
5. Results of experimental tests

The tests performed are of medium duration. The robot, driven manually by a joystick, performs simple longitudinal movements with limited rotations. The simplicity of the tests was imposed by the quality and resolution of the inertial sensor, which was found to be very noisy. Even if using sensors with higher resolution and higher cost, the algorithm is valid for long-term tests and trajectories of any kind. The preliminary experiments with the robot stopped also found the presence of a signal offset for all three measurement channels of the inertial sensor. For this reason, it was decided to carry out all the tests by keeping the robot in stasis for the first 10 seconds from the start of data recording. Subsequently, during the pre-processing of the inertial data, the used algorithm calculates the average of the values recorded in the first 10 seconds by all 3 channels. These averages constitute the offset to be subtracted from the data collected for the entire duration of the test. The results of the experimental tests are similar; below is an example for demonstrating the efficiency of the proposed innovative approach. The proposed results refer to a test carried out in the Kyutech courtyard (shown in Figure 6), in which the robot, manually guided by a joystick, after the first 10 seconds in which it was stationary in the initial position, moved with a straightforward movement for 10 seconds, covering about 4 meters, and then stopped in the final position. For this test, the accelerometric data acquired by the inertial sensor IMU in the 3 directions x, y, and z are shown in Figure 10 a; their elaborations, using numerical integration and elimination of the offset, provide the displacement data shown in Figure 10b.

It is evident that the poor quality of the inertial sensor makes the data excessively noisy, so the trajectory determined on the 3 axes differs considerably from the actual behavior, in which the robot is stationary for the first 10 seconds and then moves about 4 meters in the x-direction for the next 10, stopping definitively at the 20th second of the test.

The results obtained using the proposed method of assessing the position are much more interesting and accurate, identifying, through the data obtained from LiDAR, a landmark at the intersection of identified reference walls. The LiDAR in the experimental tests automatically acquires with a frequency of 5 Hz, and all the frames are processed as shown in section 4.2, to evaluate, through the fixed position of the landmark, the position of the robot during the test.

a)



b)

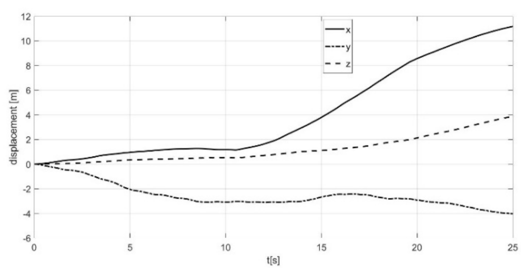


Fig. 10. a) accelerations of the IMU sensor during the Test b) displacements integrating the IMU data

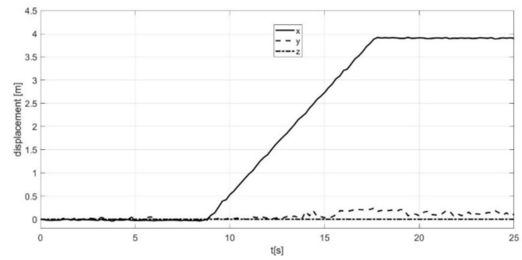
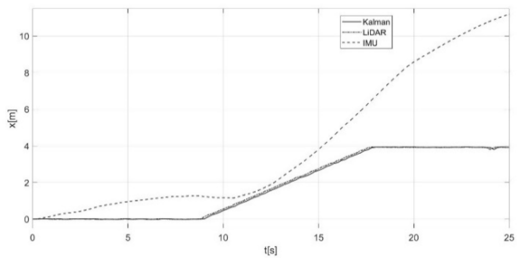


Fig. 11. Displacement estimated through the explained procedures from LiDAR frames

In this application, the odometric data obtained from the LiDAR (shown in Fig. 11) are, for all 3 coordinates x, y, and z, much closer to the real values. For this reason, the choice of the position data obtained from the LiDAR as reference (corrective) data for the Kalman filter was considered by choice of the covariance matrices R and Q (9-10), giving greater weight to the measurements made with the LiDAR. The output provided by the Kalman filter, as introduced in this work, consists of the laying of the robot as a function of time starting from the inertial controls of the robot, corrected by the observations made by the LiDAR with a frequency of about 5 Hz. The x and y coordinates estimated from the proposed approach for the considered experimental test are shown in Figure 12a and 12b, demonstrating how the Kalman filter can clean the noisy and inaccurate behavior of the IMU data. In Figures 13 and 14, a 3-dimensional view of the estimated trajectory of the robot by using the proposed procedure is shown.

a)



b)

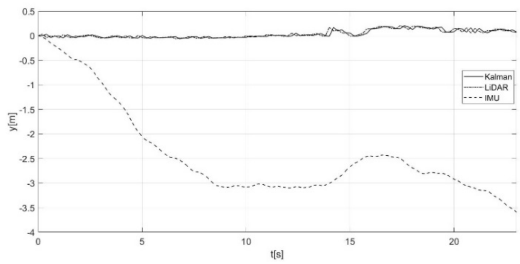


Fig. 12. a) x-displacement from IMU, LiDAR and Kalman b) y-displacement from IMU, LiDAR and Kalman

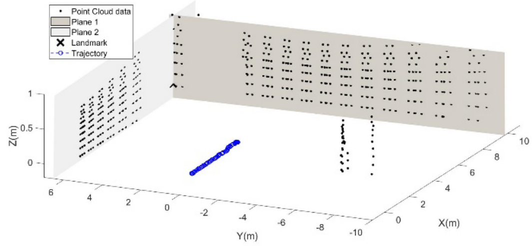


Fig. 13. Estimated robot trajectory by using the proposed approach

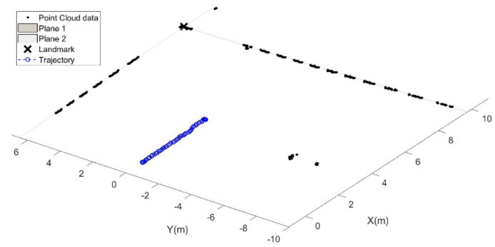


Fig. 14. Estimated robot trajectory by using the proposed approach, top view

Other experimental tests, in addition to the test whose results are shown in Figures 10–14, have been carried out by successfully estimating the trajectory with the proposed procedure. All the tests were similar to the proposed one, all conducted on a flat surface (the courtyard floor shown earlier), with simple trajectories, also including rotations, which are easy to compare with the estimated ones. From the experimental tests carried out and the results obtained we can conclude that the proposed methodology has been experimentally tested on simple paths with positive results. It can also be foreseen that, by appropriately managing the number of clusters to be identified and using more accurate inertial sensors, it is also possible to detect the position relative to detected objects, both stationary and in motion, and even to establish their trajectory and their movement concerning the robot in exploration.

6. Conclusion

Today, odometric techniques are of great importance in the world of robotics and automation; for this reason, an innovative algorithm has been proposed that can estimate the pose of the robot during inspection operations. This algorithm exploits the potential of the Kalman filter by combining inertial data with those of the LiDAR. In this paper, a modularized driving-wheels robot that is able to balance the mobile robot's efficiency according to the environment has been considered together with an efficient algorithm using LiDAR data for robot self-localization and detection of mobile targets.

The developed robot, which has been constructed by combining modules, may travel through several environments with saving energy by changing the number and arrangement of the wheels according to the environment.

Experimental tests were carried out with the robot to inspect the environments and receive the data necessary for the estimate of the odometry. From the calibration tests, it was possible to note how the IMU model installed on the robot was not suitable for estimating the installation as it was too noisy. Besides, the double integration of inertial data caused the error to grow beyond an acceptable threshold. For these reasons, in the use of the Kalman filter, greater weight was given to the filter correction phase using the LiDAR observations compared to the prediction phase, because the IMU model installed on the robot was not suitable for estimating the installation due to the noise. With this foresight, the proposed algorithm can accurately estimate the trajectory of the rover.

The particularity of the proposed approach is related to the simplicity of the procedure, that uses known tools, so it could be easily implemented, and uses a specific target (flat vertical fixed planes) for simplifying the trajectory estimation. The experimental results shown in this paper demonstrate that, in compatible scenarios, the proposed algorithm can accurately estimate the trajectory of the rover, using a Kalman filter that can compensate for the difference in accuracy between the LiDAR and the IMU data, evaluated by an initial calibration.

Besides, a specific algorithm involving the use of a clustering technique for automatically analyzing the LiDAR data has been presented and tested, demonstrating that, in specific structure scenarios, the robot can self-localize its position compared to fixed landmarks.

ACKNOWLEDGEMENTS

The authors thank the students Alessio 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, ivan.giannoccaro@unisalento.it.

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

Aimè Lay-Ekuakille – Department of Innovation Engineering, University of Salento, Lecce, 73100, Italy, aime.lay.ekuakille@unisalento.it.

Ramiro Velazquez – Universidad Panamericana, Aguascalientes, Ags, 20290, MEXICO, rvelazquez@up.edu.mx.

Paolo Visconti – Department of Innovation Engineering, University of Salento, Lecce, 73100, Italy, paolo.visconti@unisalento.it.

*Corresponding author

REFERENCES

[1] T. Goelles, B. Schlager, S. Muckenhuber, "Fault Detection, Isolation, Identification and Recovery (FDIIR) Methods for automotive perceptions sensors including a detailed literature survey for lidar", *Sensors*, vol. 20, 3662, 2020, pp. 1–21.

[2] T. Raj, F. Hanim Hashim, A. Baseri Huddin, M. Faisal Ibrahim, A. Hussain, "A survey on LiDAR scanning mechanisms", *Electronics*, vol. 741, 9, 2020, pp. 1–25.

[3] C. Debuenne, D. Vivet, "A review of visual-Lidar fusion based simultaneous localisation and mapping", *Sensors*, vol. 20, 2068, 2020, pp. 1–20.

[4] P. Forsman, A. Halme, "3-D mapping of natural environments with trees by means of mobile perception", *IEEE Transactions on Robotics*, vol. 21, 2005, pp. 482–490. 10.1109/TRO.2004.838003

[5] T. Tsubouchi, A. Asano, T. Mochizuki, S. Kondou, K.Shiozawa, M. Matsumoto, S. Tomimura, S. Nakanishi, A. Mochizuki, Y. Chiba, K. Sasaki, T. Hayami, "Forest 3D Mapping and Tree Sizes Measurement for Forest Management Based on Sensing Technology for Mobile Robots," Berlin, Heidelberg: *Springer Berlin Heidelberg*, 2014, pp. 357–368.

[6] J. Billingsley, A. Visala, M. Dunn, "Robotics in agriculture and forestry" *Springer handbook of robotics*; Springer, 2008, pp. 1065–1077.

[7] X. Liang, P. Litkey, J. Hyyppa, H. Kaartinen, M. Vastaranta, M. Holopainen, "Automatic Stem Mapping Using Single-Scan Terrestrial Laser Scanning", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, 2012, pp. 661–670. 10.1109/TGRS.2011.2161613

[8] M. A. Juman, Y. W. Wong, R. K. Rajkumar, L. J. Goh, "A novel tree trunk detection method for oil-palm plantation navigation", *Computers and Electronics in Agriculture*, vol. 128, 2016, pp. 172–180. 10.1016/j.compag.2016.09.002

[9] S. Li, H. Feng, K. Chen, K. Chen, J. Lin, L. Chou, "Auto-maps generation through self path generation in ROS based Robot Navigation", *Journal of Applied Science and Engineering*, vol. 21, no. 3, 2018, pp. 351–360.

[10] M. Ocando, N. Certad, S. Alvarado, A. Terrones, "Autonomous 2D SLAM and 3D mapping of an

environment using a single 2D Lidar and ROS", *Proc. of 2017 Latin American Robotics Symposium*, 2017.

[11] Y. Wang, C. Peng, A. Ravankar, A. Ravankar, "A single LIDAR-Based feature fusion indoor localisation algorithm", *Sensors*, vol. 1294, 2018, pp. 1–19.

[12] A. Lay-Ekuakille, A. Trotta, "Binomial Filtering to Improve Backscattered Lidar Signal Content", *Proc. of XVII IMEKO World Congress*, June 22–27, 2003, Dubrovnik, Croatia.

[13] N. I. Giannoccaro, T. Nishida, "The Design, Fabrication and Preliminary Testing of a Variable Configuration Mobile Robot", *International Journal of Robotics and Automation Technology*, 2019, 6, pp. 47–54.

[14] N.I. Giannoccaro, T. Nishida, "Analysis of the surrounding environment using an innovative algorithm based on lidar data on a modular mobile robot", *Journal of Automation, Mobile Robotics and Intelligent Systems*, vol. 14, no. 4, 2020.

[15] N.I. Giannoccaro, L. Spedicato, C. Di Castri, "A new strategy for spatial reconstruction of orthogonal planes using a rotating array of ultrasonic sensor" *IEEE Sensors*, vol. 12, no. 5, 2012, pp. 1307–1316.

[16] N.I. Giannoccaro, L. Spedicato, "Exploratory data analysis for robot perception of room environments by means of an in-air sonar scanner", *Ultrasonics*, vol. 53, no. 6, 2013, pp. 1163–1173.

[17] N.I. Giannoccaro, L. Spedicato, L. Aiello, "Kernel PCA and approximate pre-images to extract the closest ultrasonic arc from the scanning of indoor specular environments", *Measurement: Journal of the International Measurement Confederation*, vol. 58, no. 1, pp. 2014, 46–60.

[18] W. Farag, "Real-Time Autonomous Vehicle Localization Based on Particle and Unscented Kalman Filters", *Journal of Control, Automation and Electrical Systems*, vol. 32, no. 2, 2021, pp. 309–325.

[19] W.Xu, F. Zhang, "FAST-LIO: A fast, robust, LiDAR--inertial odometry package by tightly-coupled iterated Kalman-filter", *IEEE Robotics and Automation Letters,* vol. 6, no. 2, 2021, pp. 3317–3324.

[20] P. Chauchat, A. Burrau, S. Bonnabel, "Factor Graph-Based Smoothing without Matrix Inversion for Highly Precise Localization", *IEEE Trans. On Control System Technology,* vol. 29, no. 3, 2021, pp. 1219–1232.

[21] S. Muro, I. Yoshida, M. Hashimoto, T. Takahashi, "Moving-object detection and tracking by scanning LiDAR mounted on motorcycle based on dynamic background subtraction", *Artificial Life and Robotics,* Vol. 26, no. 4, 2021, pp. 412–422.

[22] H. Zhang, N. Chen, Z. Dai, G. Fan." A Multi-level Data Fusion Localization Algorithm for SLAM", *Jiqiren/Robot*, vol. 43, no. 6, 2021, pp. 641–652.

[23] J. Zhang, L. Xu, C. Bao, "An Adaptive Pose Fusion Method for Indoor Map Construction", *International Journal of Geo-Information*, vol. 10, 2021, pp. 1–22.

[24] M. Morita, T. Nishida, Y. Arita M., Shige-eda, E. di Maria, R. Gallone, N.I. Giannoccaro, "Development of Robot for 3D Measurement of Forest Environment", *Journal of Robotics and Mechatronics*, vol. 30, 2018, pp. 145–154. 10.20965/jrm.2018.p0145.

[25] https://hokuyo-usa.com/application/files/7815/9111/2405/YVT-35LX-FK_Specifications.pdf

[26] https://www.invensense.com/products/motion-tracking/6-axis/mpu-6500/

[27] https://www.cytron.io/p-10amp-5v-30v-dc--motor-driver-2-channels.

[28] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng, "ROS: an open-source Robot Operating System", *Proc. of ICRA workshop on open source software*, Kobe, Japan, vol. 3, no. 5, 2009.

[29] https://www.vizrt.com/en/products/viz-virtual-studio

[30] Matlab, "Computer Vision Toolbox", *The Mathworks*, 2018.