



Politecnico di Bari

Repository Istituzionale dei Prodotti della Ricerca del Politecnico di Bari

An RGB-D multi-view perspective for autonomous agricultural robots

This is a post print of the following article

Original Citation:

An RGB-D multi-view perspective for autonomous agricultural robots / Vulpi, Fabio; Marani, Roberto; Petitti, Antonio; Reina, Giulio; Milella, Annalisa. - In: COMPUTERS AND ELECTRONICS IN AGRICULTURE. - ISSN 0168-1699. - STAMPA. - 202:(2022). [10.1016/j.compag.2022.107419]

Availability:

This version is available at <http://hdl.handle.net/11589/249927> since: 2026-04-07

Published version

DOI:10.1016/j.compag.2022.107419

Publisher:

Terms of use:

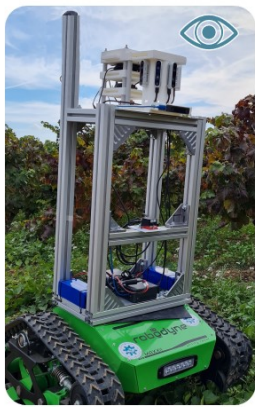
(Article begins on next page)

1 Graphical Abstract

2 An RGB-D multi-view perspective for autonomous agricultural
3 robots

4 Fabio Vulpi, Roberto Marani, Antonio Petitti, Giulio Reina, Annalisa Milella

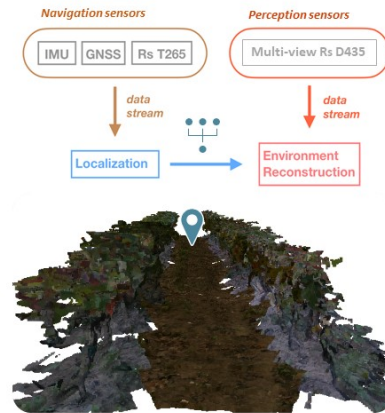
An RGB-D multi-view perspective for autonomous agricultural robots



Develop a cost effective multi-view RGB-D sensing system for advanced robot perception in agricultural environments

Fuse localization data with an EIF for accurate 6-DOF vehicle position estimation

Use localization estimates to merge point clouds in a geo-referenced 3D map of the traversed crop



5 Highlights

6 **An RGB-D multi-view perspective for autonomous agricultural**
7 **robots**

8 Fabio Vulpi, Roberto Marani, Antonio Petitti, Giulio Reina, Annalisa Milella

- 9 • Multi-view RGB-D system for proximal sensing of high-value crops on
10 a narrow scale
- 11 • Combination of visual and localization data to produce high-resolution
12 3D maps
- 13 • A robotic system for efficient in-field data gathering for crop monitoring
14 tasks

15 An RGB-D multi-view perspective for autonomous
16 agricultural robots

17 Fabio Vulpi^{a,b}, Roberto Marani^a, Antonio Petitti^a, Giulio Reina^b, Annalisa
18 Milella^a

^aInstitute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing (STIMA), Institute of National Research Council of Italy (CNR), via Amendola 122 D-O, Bari, 70126, Bari, Italy

^bDepartment of Mechanics, Mathematics and Management, Polytechnic of Bari, Via Orabona 4, Bari, 70125, Bari, Italy

19 **Abstract**

Automated in-field data gathering is essential for crop monitoring and management and for precision farming treatments. To this end, consumer-grade digital cameras have been shown to offer a flexible and affordable sensing solution. This paper describes the integration and development of a cost-effective multi-view RGB-D device for sensing and modelling of agricultural environments. The system features three RGB-D sensors, arranged to cover a horizontal field of view of about 130 deg in front of the vehicle, and a suite of localization sensors consisting of a tracking camera, an RTK-GPS sensor and an IMU device. The system is intended to be mounted on-board an agricultural vehicle to provide multi-channel information of the surveyed scene including color, infrared and depth images, which are then combined with localization data to build a multi-view 3D geo-referenced map of the traversed crop. The experimental demonstrator of the multi-sensor system is presented along with the steps for the integration of the different sensor data into a unique multi-view map. Results of field experiments conducted in a commercial vineyard are included, as well, showing the effectiveness of the proposed system. The resulting map could be useful for precision agriculture applications, including crop health monitoring, and to support autonomous driving.

20 *Keywords:* Agricultural robotics, advanced perception, RGB-D sensing,
21 multi-view imaging, environment mapping, precision farming

22 1. Introduction

23 Persistent and timely crop monitoring is crucial for the development of
24 sustainable food production systems. While remote sensing from satellites
25 and aircrafts has been successfully used for some decades to allow for rapidly
26 mapping and characterize wide areas through acquisition of multispectral im-
27 agery and 3D data, they generally lack the spatio-temporal resolution needed
28 for precision agriculture or phenotyping tasks. In crops with smaller exten-
29 sion, Unmanned Aerial Vehicles (UAVs) have been effectively adopted for
30 remote crop survey with higher spatial and temporal resolution compared
31 to satellite and airborne devices. In high-density crops, however, collecting
32 information on biophysical properties at plant or leaf/fruit level via aerial
33 sensing may be still infeasible. As a result, ground-based sensing through
34 Unmanned Ground Vehicles (UGVs) has been proposed for in-field close-
35 range applications [1].

36 In this respect, imaging sensors embedded on ground vehicles have recently
37 attracted much attention as an effective solution to collect high resolution
38 proximal data and provide information on plant characteristics at centimetre
39 or sub-centimetre scale. At the same time, they can be used for vehicle guid-
40 ance automation and scene perception and understanding in general, thus
41 contributing to all aspects of crop monitoring automation, from data gath-
42 ering up to high-level data processing and interpretation.

43 Among imaging sensors, consumer-grade RGB-D cameras, i.e., sensors that
44 embed color and depth sensing into a unique device, have emerged in the
45 last years in mobile robotics applications, to provide the vehicle with a real-
46 time representation of both appearance and 3D structure of the environment.
47 One shortcoming of typical consumer-grade color-depth sensors is their lim-
48 ited field-of-view, which makes them unsuitable for applications that need
49 continuous survey of extensive areas, like in agricultural settings.

50 In this work, a multi-view RGB-D camera system is proposed to enhance the
51 perception ability of an unmanned agricultural robot. The system features
52 three RGB-D cameras arranged to cover a horizontal field of view of about
53 130 deg. The cameras are integrated with localization sensors, including
54 a stereo-based tracking camera, an RTK-GPS and an IMU, which provide
55 accurate vehicle position information for image geo-referencing based on Ex-
56 tended Information Filter (EIF). All the sensors are physically integrated in
57 a custom-built sensor box, which is designed to be self-contained from both a
58 computational and energy point of view and independent from the particular

59 vehicle architecture. The system is intended to generate accurate maps to
60 facilitate operations on a narrow scale with a smaller environment footprint.
61 To this end, a point cloud assembler that uses EIF-based pose estimates and
62 the known relative poses between sensors is developed to reconstruct a geo-
63 referenced multi-view 3-D map of the traversed environment, which could
64 provide a useful input to precision farming technologies and crop monitoring
65 tasks.

66 An additional use of the map is that it can be incorporated into a high-fidelity
67 simulator that can support the development and pre-testing of algorithms for
68 autonomous driving. In this respect, the map can be converted into a mesh
69 representation using the Ball Pivoting Algorithm (BPA) and imported into
70 a simulation environment under Gazebo [2].

71 The rest of the paper is structured as follows. First, related work is pre-
72 sented in Section 2. The hardware and software design of the multi-view
73 sensing device is reported in Section 3. Section 4 describes the multi-view
74 3D mapping approach. The simulator is detailed in Section 5. Experimen-
75 tal results obtained in real agricultural settings are presented in Section 6.
76 Finally, conclusions are drawn in Section 7.

77 2. Related Work

78 The availability of up-to-date and accurate data is an essential pre-requisite
79 for precision farming tasks, such as variable rate application of fertilizers/pesticides,
80 identification of infected plants or invasive species, and controlled traffic farm-
81 ing. While satellite and airborne technologies have been in use for some
82 decades to effectively provide multi-spectral and 3D information in wide
83 agricultural and forestry areas, these platforms generally lack the resolu-
84 tion needed to observe stems, leaves or fruits. Satellite images typically have
85 pixel resolution of hundreds of meters and airborne sensing may provide res-
86 olution of a few meters, whereas monitoring orchards or vineyards requires
87 observations at a smaller scale. Information update frequency is also limited,
88 varying from hours to several days. In crops with smaller extension, UAVs
89 equipped with RGB, multispectral or LiDAR sensors, have been adopted to
90 overcome these bottlenecks, allowing for efficient crop survey at user-defined
91 spatio-temporal resolutions to assess vegetation vigor or for canopy charac-
92 terization [3], [4]. However, in high density crops, using aerial data can still
93 be ineffective for precise measurement at leaf/fruit level, e.g., for health sta-
94 tus assessment and yield estimation.

95 As an alternative or complementary approach, proximal sensing from ground-
96 based or manually deployed devices can be performed. Proximal sensors
97 range from RGB cameras to high-resolution hyperspectral imaging, infrared
98 (IR) thermal cameras, and 2D/3D LiDARs. Applications include fruit de-
99 tection and counting [5], up to plant phenotyping [6], health status assess-
100 ment and growth monitoring [7], [8]. While these methods were proved to
101 be effective and accurate for detailed information extraction, they are of-
102 ten constrained to structured environments, such as greenhouses and specific
103 acquisition conditions, such as controlled illumination or pre-defined posi-
104 tioning of the sensing devices, or they require the adoption of expensive
105 high-resolution sensors [9], which limits their practical implementation.
106 In order to address these issues, crop monitoring by agricultural ground
107 robots has been proposed as a step forward to automated proximal mea-
108 surement and characterization of high-value crops and soils [10], [11]. While
109 much work has been done in the context of ground robots for harvesting and
110 picking operations, the use of UGVs for in-field crop monitoring and assess-
111 ment has been proposed more recently. UGVs can carry a number of sensing
112 devices, thus potentially providing an efficient means to gather multi-modal
113 information at a narrow scale. At the same time, they can be equipped with
114 manipulators and actuators to perform targeted actions, such as selective
115 spraying or fertilizing, with relatively high operating times.
116 Although UGVs offer enough payload to transport a number of bulky sen-
117 sors, keeping low complexity and costs is a major requirement for in-field
118 implementation. In this respect, visual sensors mounted on ground robots
119 have been shown to provide an efficient and affordable solution in a wide
120 range of agricultural applications, including plant and fruit detection, fruit
121 grading, ripeness detection, yield prediction, plant and fruit health protec-
122 tion and disease detection. In addition, visual sensors provide a rich source
123 of information to support autonomous navigation functions such as localiza-
124 tion, obstacle detection and situation awareness in general [1], [12].
125 Among visual sensors, portable consumer-grade RGB-D cameras, like Mi-
126 crosoft Kinect, have been receiving growing attention, as an effective means
127 to recover in real-time 3D textured models of plants and extract plant and
128 fruit features [13], although the application of this sensor remains mostly lim-
129 ited to indoor contexts. A novel family of highly portable, consumer depth
130 cameras has been introduced by Intel in 2015 (R200 and D4xx, Santa Clara,
131 CA, USA). These cameras are similar to the Kinect sensor in scope and cost,
132 but use a different working principle based on IR stereo, which makes them

133 more suitable for outdoor conditions. In addition, their output include RGB
134 information, infrared images and 3D depth data, thus covering a wide range
135 of information about the scene. The potential of these sensors for agricul-
136 tural applications has been investigated in recent works [14], [15].
137 Following this research trend, this work explores the potential of a multi-view
138 RGB-D system for geo-referenced image acquisition and mapping of a high-
139 value crop, like a vineyard. The device is built following a modular approach
140 and can be mounted on any agricultural vehicle to provide ground-based 3D
141 reconstruction of the traversed crop rows. Data acquisition and processing
142 can be carried out during vehicle operations, in a non-invasive and completely
143 automatic way, while requiring low investment and maintenance costs.
144 One specific aspect addressed is accurate vehicle localization. Localization
145 of the UGV is essential for correct merging of point cloud streams and thus
146 for the construction of geo-referenced 3-D maps. In [16], the UGV pose
147 estimation problem is formulated as a pose graph optimization to mitigate
148 sensor drift and significantly improve state estimation accuracy using a Digi-
149 tal Elevation Model (DEM) and a Markov Random Field (MRF) assumption.
150 Authors in [17] proposed a Simultaneous Localization And Mapping (SLAM)
151 method for generating the map of an agricultural environment and simulated
152 it on Gazebo and Robot Operating System (ROS) for the case of an apple
153 farm, showing good results in fruit mapping. A well-established solution to
154 the localization problem to fuse information from multiple sensors is Kalman
155 filtering. In this work, we use the information form of the Kalman Filter as
156 data fusion strategy for heterogeneous sensors. The reason is related to the
157 high reliability of such algorithm, as confirmed by recent research (e.g.,[18],
158 [19]). Other alternatives have been investigated in the literature, including,
159 for example, particle filtering, which however has proven to be less accurate
160 for localization purposes ([20], [21]).

161 **3. Multi-view Sensing Device**

162 This section describes the development of a multi-sensor box for close
163 range sensing and modelling of agricultural environments. The sensor suite
164 is intended to be mounted on board an agricultural robot and is designed to
165 be self-contained, both from a computational and energy point of view, and
166 independent from the particular vehicle architecture.

167 3.1. Hardware Design

168 The sensor suite is shown in Figure 1 (a). It consists of two sensor arrays,
169 namely a *Perception Sensors* array and a *Navigation Sensors* array. The
170 perception sensors include three Intel RealSense D435 RGB-D cameras ar-
171 ranged to cover a wide horizontal field of view of about 130 deg in front of the
172 vehicle, which extends up to about 145 deg when considering infrared depth
173 information only. The mounting case allows one to alternatively place up to
174 two cameras in lateral configuration, e.g., to keep the image plane parallel
175 to a crop row for tasks such as row following and/or monitoring. A closeup
176 of the multi-camera system is shown in Figure 1 (b). The navigation sen-
177 sors comprise one Intel RealSense tracking camera T265, one X-Sense IMU
178 MTi-300 and two U-Blox GPS Zed-F9P providing RTK-GPS data in rover-
179 base configuration. All sensors are integrated in a 3-D printed PLA box (see
180 Figure 1 (c)), which was designed following a modular approach, so that it
181 can be assembled in multiple ways according to the specific needs of the test
182 field. The described sensor suite can be fixed to the vehicle through a metal
183 frame, built with aluminium bars and plates and designed to be stable and
184 of adjustable height. Two Intel NUC7i7DNHE computers are used for data
185 gathering. The PCs, powered by lithium batteries, are fixed at the bottom
186 of the metal frame. Overall, the proposed sensor box provides a flexible and
187 self-contained data gathering device with a cost of about 6.5k €(i.e., 27% for
188 the two processing units, 45% for the IMU, 14% for the cameras, 7% for the
189 GPS, and 7% for the batteries).

190 3.2. Acquisition Software Design

191 The data gathering pipeline of the sensor suite is shown in Figure 2. The
192 sensor box provides two processing units, one running Ubuntu and the other
193 running Windows 11. The NUC Ubuntu is devoted to gathering positional
194 measurements produced by the GPS sensor and the IMU sensor. Data ac-
195 quisition is made through ROS drivers using a dedicated ROS node for each
196 sensor. Then, all the acquired data are stored in ROS bags.
197 For image acquisition and storage, a software package, named SensorBox,
198 was developed using the Intel RealSense SDK 2.0 (v. 2.49), running on
199 NUC Windows. The software architecture of the whole package is divided
200 into two executables: *MultiBagReader* and *MultiBagWriter*. The scheme of
201 the first executable (*MultiBagWriter*) is shown in Figure 3. It works in a
202 producer-consumer logic, where the Intel RealSense cameras connected to the
203 processing unit are first opened to produce the data which is then consumed,

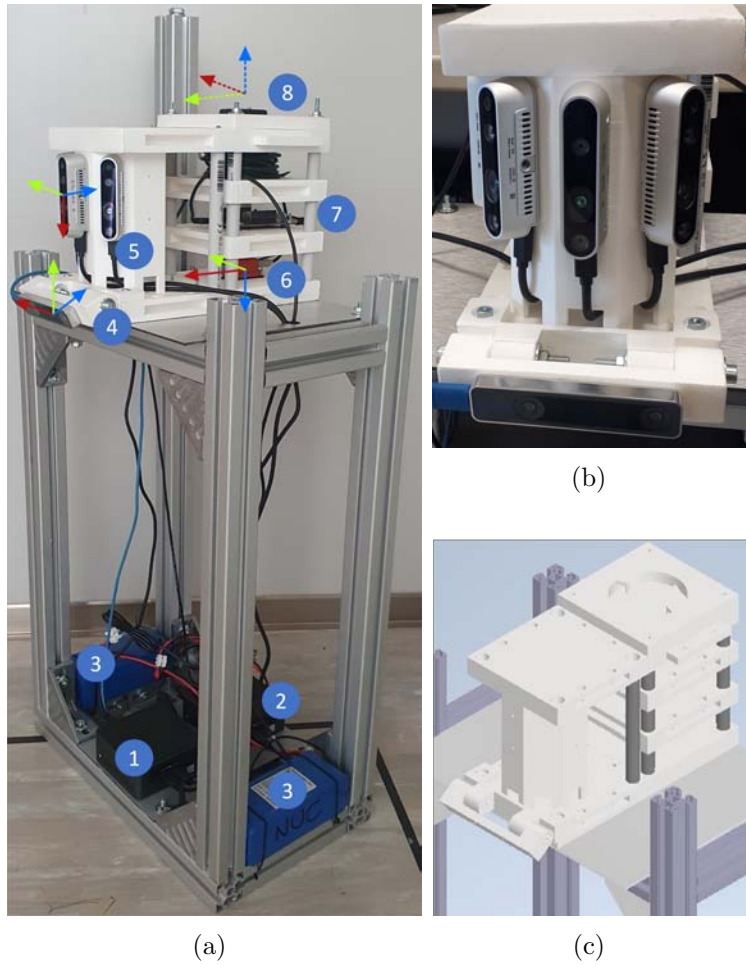


Figure 1: (a) Demonstrator of the UGV's sensor box: (1)-(2) Intel NUC Windows PCs; (3) Batteries, (4) T265 Camera, (5) D435 Cameras, (6) X-Sense MTI-300 IMU, (7) U-blox ZED F9P board, (8) Sensor Box GPS antenna. (b) Closeup of the multi-camera system. (c) CAD model of the sensor frame.

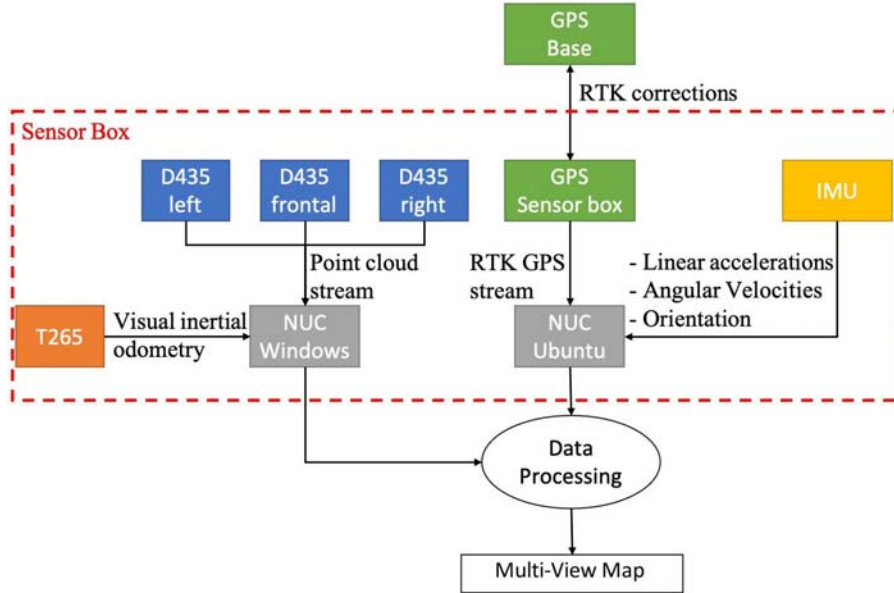


Figure 2: Sensor box data flow.

204 i.e. displayed, to show the acquired field of view and/or the computed visual
 205 odometry. Then, when the user starts the acquisition, the data are encapsu-
 206 lated in several ROS bags, stored on the local hard disk of the NUC. In this
 207 way, each camera produces a bag file at the maximum achievable rate (up to
 208 30 fps), without any further processing to prevent frame drops. It is worth
 209 noticing that each camera works in a free run mode and, thus, their frames
 210 are not temporally synchronized, i.e. acquired exactly at the same time in-
 211 stant. The second executable (*MultiBagReader*) opens the bags, divides the
 212 RealSense pipelines to have single streams in each pipeline, and then reports
 213 all the acquired frames to a global temporal reference, thus performing soft-
 214 ware synchronization. The software features a user interface for both writing
 215 and reading modules, as shown in Figure 4. The open-source code of the
 216 software is available on GitHub (<https://github.com/ispstiima/SensorBox>).
 217

218 3.3. Sensor Synchronization and Calibration

219 The association of heterogeneous data requires temporal and spatial cali-
 220 bration. For time synchronization, a timestamp-based approach was adopted,
 221 whereby each sensor observation was marked with a timestamp. In addition,

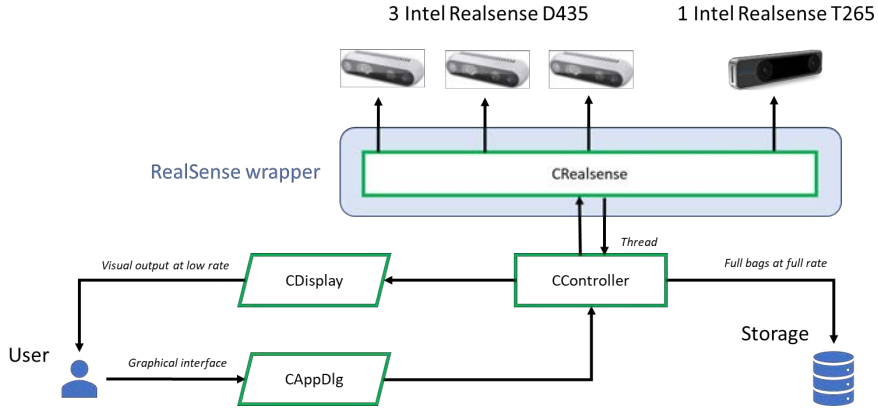


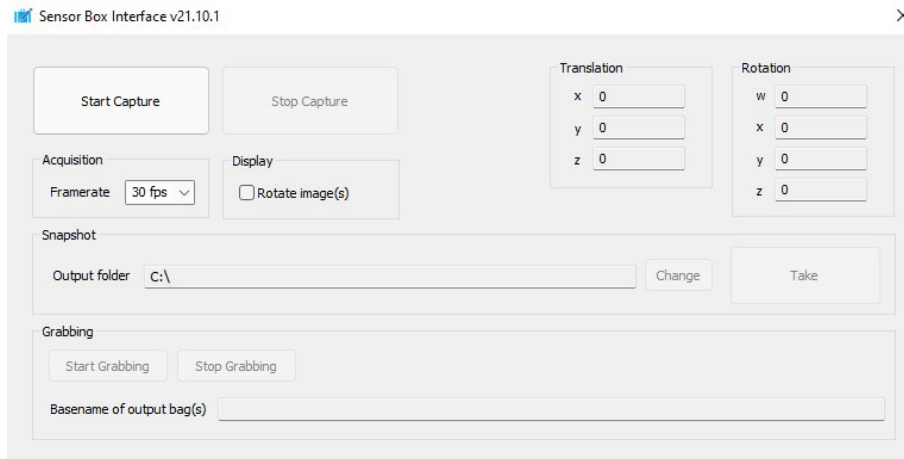
Figure 3: Schematic of the image acquisition software.

222 to register all sensor data with respect to a common reference frame, spatial
 223 calibration was performed to estimate the relative position and orientation of
 224 the sensors with respect to each other. Spatial calibration was performed by
 225 construction, considering that all the sensors are located in the sensor box
 226 at fixed positions. This proved to be sufficiently accurate for the purpose
 227 of this work, although optimization strategies, such as the one proposed by
 228 the authors in [22], can be also adopted to further improve the registration
 229 accuracy.

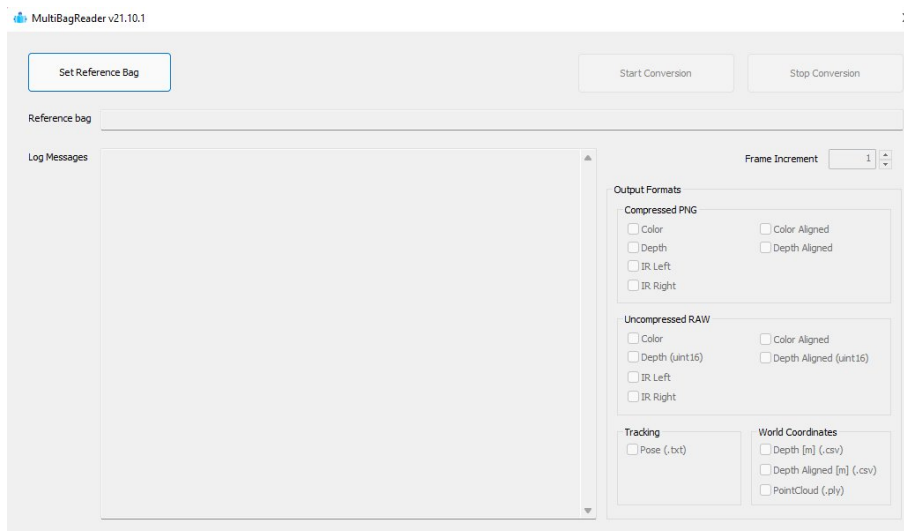
230 4. Multi-view 3D Mapping

231 The data acquired by the sensor suite are processed to build a multi-view
 232 map of the traversed environment, following multiple stages. First, data from
 233 GPS, IMU and T265 sensors are fused by an EIF to generate pose estimates.
 234 Successively, the point clouds obtained by each of the three RGB-D cameras
 235 are assembled into a unique map, using the EIF pose estimates and the
 236 known relative poses between the sensors. The map can be then converted
 237 into a 3D mesh representation for efficient storage and inspection, as well as,
 238 for import in a dedicated simulation environment, as will be described later
 239 in Section 5. In more detail, with reference to Figure 1, let us introduce the
 240 reference frames denoted with the following subscripts:

- 241 • *sb*: Sensor Box frame (Figure 1, a-6)
- 242 • *w*: East-North-Up world frame (Figure 1, a-8)



(a)



(b)

Figure 4: User interface of the multi-view camera system: (a) interface for data acquisition and storage (MultiBagWriter); (b) interface for reading stored image databases (MultiBagReader).

243 • *cam*: Camera Frame (Figure 1, a-5)

244 • *s*: Reference frame for the *s*-th sensor (RTK-GPS, T265, IMU).

245 Furthermore, quantities of interest are presented here to facilitate the
246 understanding of the mapping reconstruction process:

247 • $p_{B-A}(t) \in \mathbb{R}^{1,3}$: position vector of reference frame *A* at time *t* expressed
248 in reference frame *B*

249 • $\mathbf{q}_{B-A}(t) \in \mathbb{C}^{1,4}$: quaternion orientation of reference frame *A* at time *t*
250 expressed in reference frame *B*. Note that only quaternions are denoted
251 in bold.

252 For the reader's convenience, basic concepts on quaternion analysis ap-
253 pear in the Appendix. The interested reader is referred to the literature (e.g.,
254 [23]) for more details.

255 The mapping algorithm proceeds according to the following three steps:

1. *Pose estimation*: position and orientation of the sensor box can be estimated from different sensors (RTK-GPS, T265, IMU) and fused within an EIF. In general, the sensor *s* can provide information about its position and orientation expressed either in the world frame or with respect to its initial pose. In the former case, Equations (1) give the sensor box orientation \mathbf{q}_{w-sb} and position p_{w-sb} in the world frame, knowing \mathbf{q}_{s-sb} and p_{sb-s}

$$\begin{aligned} \mathbf{q}_{w-sb}(t) &= \mathbf{q}_{w-s}(t)\mathbf{q}_{s-sb} \\ [p_{w-sb}(t), 0] &= [p_{w-s}(t), 0] - \mathbf{q}_{w-sb}(t)[p_{sb-s}, 0] \end{aligned} \quad (1)$$

If the sensor provides $p_{s_0-s}(t)$ and $\mathbf{q}_{s_0-s}(t)$ with respect to its initial condition s_0 , Equations (2) compute p_{w-sb} and \mathbf{q}_{w-sb}

$$\begin{aligned} \mathbf{q}_{w-sb}(t) &= \mathbf{q}_{w-sb_0}\mathbf{q}_{sb_0-s}(t)\mathbf{q}_{s-sb} \\ [p_{w-sb}(t), 0] &= \mathbf{q}_{w-sb_0}\mathbf{q}_{sb-s}[p_{s_0-s}(t), 0]\mathbf{q}_{s-sb}\mathbf{q}_{sb_0-w} - \\ &\quad + \mathbf{q}_{w-sb}(t)[p_{sb-s}, 0] + \\ &\quad + [p_{w-sb_0}, 0] \end{aligned} \quad (2)$$

256 Apart from $\mathbf{q}_{sb-s} = \mathbf{q}_{s-sb}^{-1}$, initial position and orientation of the sensor
257 box in the world frame (p_{w-sb_0} and \mathbf{q}_{w-sb_0}) are needed to compute p_{w-sb}

258
259

and \mathbf{q}_{w-sb} . In Equations (2) notation $[p, 0]$ indicates the quaternion with zero real part corresponding to position vector p . Please note that in both Equations (1) and (2) quaternion multiplication is omitted for ease of notation. The predictive model used to implement the EIF is expressed by Equations (3)

$$\begin{aligned}\tilde{p}_{t+1|t} &= \tilde{p}_{t|t} + \tilde{v}_{t|t} \cdot \Delta t \\ [\tilde{v}_{t+1|t}, 0] &= [\tilde{v}_{t|t}, 0] + (\tilde{\mathbf{q}}_{t|t}[a_{sb}, 0]\tilde{\mathbf{q}}_{t|t}^{-1} - [g, 0]) \cdot \Delta t \\ \tilde{\mathbf{q}}_{t+1|t} &= \|(\tilde{\mathbf{q}}_{t|t} + \frac{\Delta t}{2} \cdot \tilde{\mathbf{q}}_{t|t}[\omega_{sb}, 0])\|\end{aligned}\quad (3)$$

260
261
262
263
264
265
266

where position \tilde{p} , velocity \tilde{v} , and quaternion $\tilde{\mathbf{q}}$ describe the state of the sensor box over time expressed in the w frame. Subscript $t + 1|t$ denotes predicted value and $t|t$ indicates posterior value corrected with measurements. Expression $\|\mathbf{q}\|$ represents quaternion normalization. Angular velocity ω_{sb} and linear acceleration a_{sb} are measurements provided by the IMU and expressed in the sb frame, and g denotes gravity acceleration in the world frame.

2. *Point cloud assembly* The pose estimates are then used to transform and assemble the point clouds provided by each RGB-D camera in its own frame. Denoting with $P_{cam} \in \mathbb{R}^{3,1}$ a generic point of the point cloud in the camera frame cam and with $T_{cam} \in \mathbb{R}^{4,4}$ the homogeneous transformation matrix from frame cam to frame sb , Equation (4) expresses P_{cam} in the sb frame as P_{sb} .

$$[P_{sb}^T, 1]^T = T_{cam}[P_{cam}^T, 1]^T \quad (4)$$

The homogeneous transformation matrix T_{sb} from the sensor box frame to the inertial frame can be obtained by Equation (5) using the pose estimated by the EIF

$$T_{sb} = \begin{bmatrix} rotm(\tilde{\mathbf{q}}_{t|t}) & \tilde{p}_{t|t}^T \\ [0, 0, 0] & 1 \end{bmatrix} \quad (5)$$

where $rotm(\mathbf{q}) \in \mathbb{R}^{3,3}$ returns the rotation matrix uniquely assigned to quaternion \mathbf{q} . Finally, 3D points can be transformed from the sb to the w frame using Equation (6)

$$[P_w^T, 1]^T = T_{sb}[P_{sb}^T, 1]^T \quad (6)$$

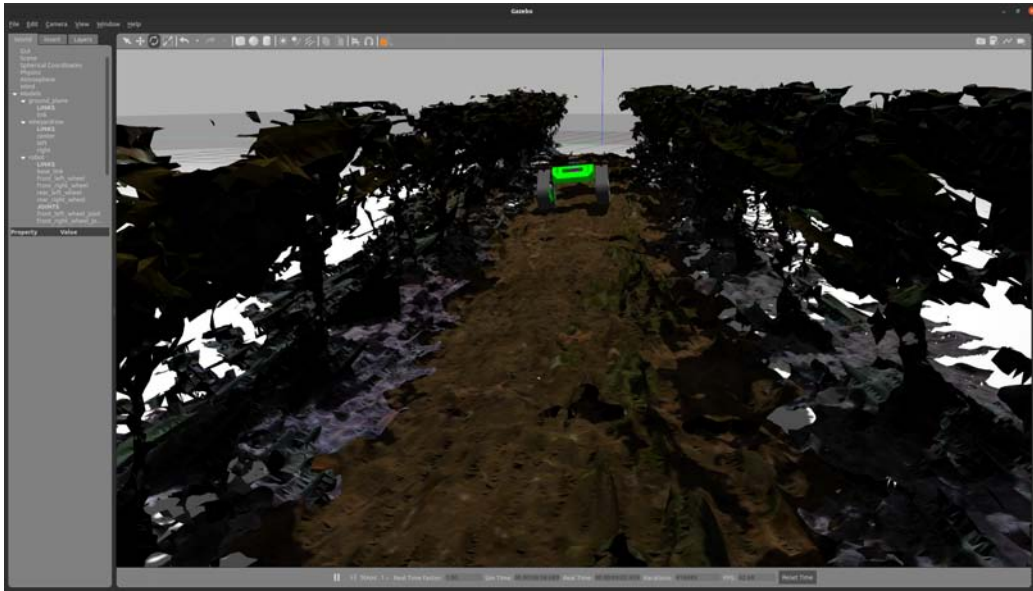


Figure 5: Gazebo simulation environment composed of a vineyard row and a mobile robot.

267 3. *Map merging.* Assembled point clouds gathered by the three cam-
 268 eras and transformed in the world frame are then merged together and
 269 downsampled (voxel grid size= 0.01 m).

270 5. Simulation Environment

271 The possibility to simulate robots in outdoor environments under differ-
 272 ent conditions is of paramount importance in agricultural robotics. In this
 273 regard, the most famous simulator framework used by the robotic commu-
 274 nity is Gazebo [2]. Gazebo is an open-source simulator born in 2002 for
 275 robotics. After over 15 years of development, Gazebo is undergoing a signifi-
 276 cant upgrade and modernization. Today, Gazebo is part of Ignition, which is
 277 a collection of open-source software libraries designed by Open Robotics to
 278 simplify the development of high-performance applications. Creating Gazebo
 279 models means creating SDF (Simulation Description Format) files to define
 280 SDF Model Objects. The primary element composing an SDF file is the link.
 281 A link contains the physical properties of one body of the model. Each link
 282 may have many collision and visual elements. Usually, these two elements
 283 define the 3D mesh file describing the 3D surface of the object. Gazebo re-
 284 quires that mesh files be formatted as STL, Collada, or OBJ, with Collada

285 and OBJ preferred formats.

286 In this work, the point cloud map, obtained as described Section 4, is modeled
287 using a meshing algorithm, which allows one to generate a mesh-based repre-
288 sentation for map import in the Gazebo simulation environment, according
289 to the following steps:

- 290 • **Downsampling:** to downsample the obtained dense map is a not
291 mandatory task that enables to speed up the mesh reconstruction pro-
292 cess and improves the outcome of the whole process. To accomplish
293 this task, the samples are generated according to a Poisson-disk distri-
294 bution [24];
- 295 • **Normals computation:** the knowledge of the normals is necessary to
296 reconstruct the surface of the elements composing the map. Normals
297 are computed on the basis of the 10 closest points;
- 298 • **Surface reconstruction:** this step regards the reconstruction of the
299 surface starting from the set of points and normals. In this case, the
300 Ball Pivoting algorithm is used [25] to compute a triangle mesh. It is
301 based on the principle that three points form a triangle if a ball of a
302 user-specified radius touches them without containing any other point;
- 303 • **Texture mapping:** the texture mapping is build by triangle-by-triangle
304 parametrization;
- 305 • **Color transfer:** this step concerns the process of projecting a 2D
306 image to a 3D model's surface for texture mapping, the so called *UV*
307 *mapping*. Once a UV map is available, the color can be transferred to
308 the reconstructed surface;
- 309 • **Save the mesh:** finally, the mesh is ready to be exported in a suitable
310 format.

311 Thus, thanks to the created mesh files, it is possible to develop a Gazebo SDF
312 model object describing the reconstructed vineyard row. As an example,
313 Figure 5 showcases a mobile robot crossing a vineyard row developed by
314 following the procedure described above.



Figure 6: Robotic platform used for in-field testing equipped with the multi-sensor box.

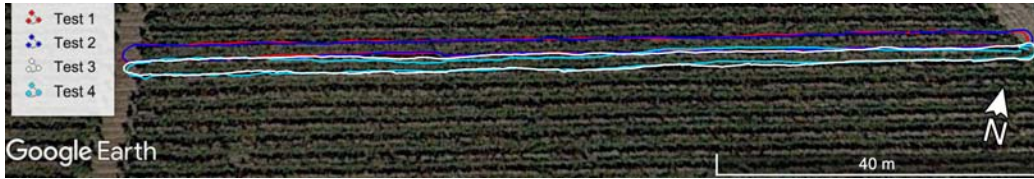


Figure 7: Google Earth view of the four paths estimated by EIF in a commercial vineyard, San Donaci, Apulia Region, Italy ($40^{\circ}27'16.2''\text{N}$ $17^{\circ}54'30.6''\text{E}$).

315 6. In-field Testing

316 The multi-sensor system is mounted and integrated on a tracked robot
 317 developed at the Politecnico of Bari, and it is tested in field conditions, as
 318 shown in Figure 6. Dedicated tests are performed in a commercial vineyard
 319 in San Donaci, Apulia region, Italy. Specifically, the robot is guided to follow
 320 closed-loop trajectories around different crop rows while gathering the sensor
 321 data. The data were then processed offline to recover the 6DoF path and the
 322 3D map of the environment.

323 In this section, first, the localization performance of the proposed system is
 324 analyzed in terms of accuracy and repeatability. Then, the mapping results
 325 are discussed.

326 6.1. Localization performance

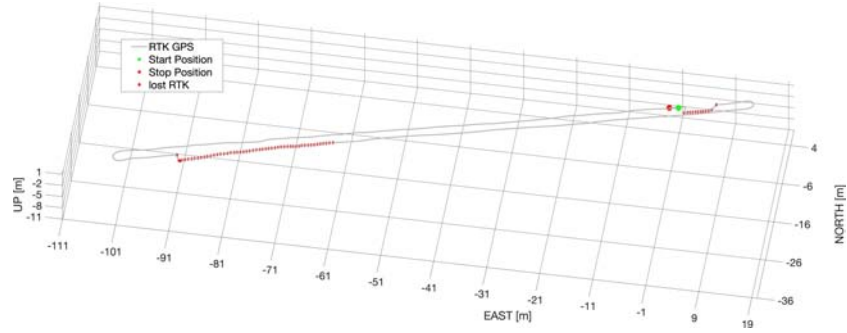
327 Four closed-loop runs are considered, performed along two different crop
 328 rows of about 120 m length, referred to as Test 1 to Test 4 in the following.
 329 They belong to two field campaigns carried out in September and Octo-
 330 ber 2021, respectively, during different times of the day. Three localization
 331 sources are compared, namely RTK-GPS only, T265 only and EIF. A pro-
 332 jection in Google Earth view of the trajectories reconstructed by EIF for the
 333 four paths is shown in Figure 7. Numerical results for all runs are collected
 334 in Table 1, showing the discrepancy in the East-North-Up (w) frame between
 335 the starting and ending points of the trajectory, expressed in terms of 3D Eu-
 336 clidean distance (D), 2D Euclidean distance in the motion plane (D_{EN}) and
 337 altitude distance (D_U), and the standard deviation of altitude measurements
 338 (σ_U) along the entire path.

339 The robot path as estimated by each localization source is reported in
 340 Figure 8 for Test 1. In this test, pose estimates using only RTK-GPS (Fig-
 341 ure 8 (a)) are consistent as long as RTK correction is available. The starting
 342 and ending points are close to each other and the altitude estimate is stable

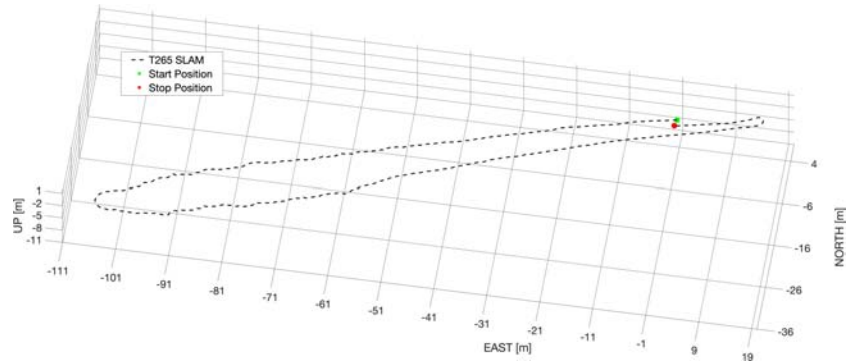
Table 1: Comparison of different localization sources along four robot paths (Test 1 to 4): discrepancy in the East-North-Up frame between the starting and ending points of the trajectory expressed in terms of 3D Euclidean distance (D), 2D Euclidean distance in the motion plane (D_{EN}), altitude distance (D_U), and standard deviation of altitude measurements (σ_U) along the entire path.

<i>Test</i>	<i>Source</i>	$D[m]$	$D_{EN}[m]$	$D_U[m]$	$\sigma_U[m]$
1	RTK-GPS	1.501	1.511	0.026	0.451
	T265	1.276	1.275	0.046	2.425
	EIF	1.514	1.513	0.032	0.228
2	RTK-GPS	0.579	0.579	0.013	0.452
	T265	5.909	4.921	3.270	2.199
	EIF	0.579	0.579	0.011	0.209
3	RTK-GPS	1.852	1.517	1.063	0.370
	T265	7.264	5.811	4.359	1.440
	EIF	1.833	1.484	1.077	0.355
4	RTK-GPS	1.290	1.073	0.717	0.670
	T265	5.341	4.760	2.424	1.497
	EIF	0.739	0.356	0.647	0.428

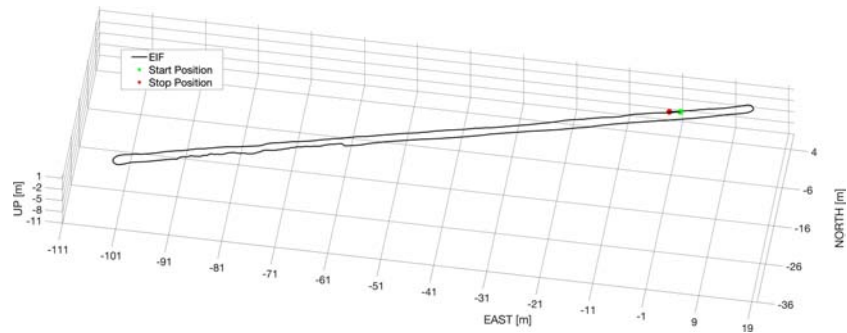
343 except when the connection to the base GPS is lost and the RTK correction
 344 is missing (red diamonds in Figure 8 (a)). Figure 8 (b) shows the path as
 345 reconstructed by the T265 proprietary visual-inertial SLAM algorithm in its
 346 own frame and successively transformed in the world frame. Compared to
 347 RTK-GPS path, the distance between starting and ending points estimated
 348 by the T265 camera is 15.6% smaller in terms of East-North coordinates
 349 (D_{EN}) but 77% larger in terms of altitude (D_U). Low accuracy of vertical
 350 displacement estimates leads to large standard deviation of altitude mea-
 351 surements (σ_U), which for T265 is of 2.43 m, about 5 times larger than the
 352 one obtained by RTK-GPS. Figure 8(c) shows the path reconstructed by the
 353 EIF. The EIF uses linear acceleration and angular velocity measures to make
 354 state predictions using the model described by equations (3), and corrects its
 355 predictions using measures of RTK-GPS (world position and velocity), IMU
 356 (world orientation) and T265 (relative position and orientation). The EIF
 357 estimates a difference between starting and ending points 0.18% larger than
 358 the RTK-GPS in terms of East-North position and 23% larger in terms of
 359 vertical displacement. However, the standard deviation of altitude measures



(a)



(b)



(c)

Figure 8: Localization results for Test 1: (a) from RTK-GPS only; (b) from T265 camera only; (c) after EIF fusion of RTK-GPS, IMU and T265 measurements. In (a), red diamonds are overlaid in two different zones without RTK coverage due to connection loss.

360 is 0.22 m for the EIF, i.e., 49% smaller than the one provided by RTK-GPS,
361 suggesting an overall improvement in position estimate when fusing mea-
362 surements with the EIF. This improvement is due to the fact that the RTK
363 corrections are not available when connection is lost with the base GPS,
364 whereas the EIF adjusts the position estimates for these instants using pre-
365 dictions with IMU data and short-term measures of the T265 camera when
366 the covariance of position and velocity provided by the GPS grows.

367 When considering a second run along the same row (Test 2), similar results
368 are obtained (see Figure 9 and the corresponding row in Table 1) in terms
369 of σ_U which attests to 0.45 m for RTK-GPS, 2.20 m for T265 camera and
370 0.21 m for EIF. The discrepancy between starting and ending points (D) for
371 T265 is higher than the one obtained from RTK-GPS indicating that visual
372 inertial odometry should be only used for short-term displacement estima-
373 tion. Again, the use of EIF allows for a reduction of σ_U of 53% with respect
374 to GPS and of 90% with respect to T265, while preserving loop closure ac-
375 curacy.

376 The localization results for a path along a different crop row (Test 3) is re-
377 ported in Figure 10. In this case, the T265 results in a substantially degraded
378 estimation, and EIF mainly relies on GPS leading to σ_U of 0.35 m. On the
379 contrary, Figure 11 refers to an example where the quality of GPS signal is
380 poor in several parts of the trajectory (Test 4). Again, EIF is able to com-
381 pensate for the GPS outages mainly relying on T265 information showing
382 better performance for all the metrics.

383 6.2. Mapping

384 For each geo-referenced position, the corresponding multi-view data can
385 be recovered. As an example, Figure 12 shows the robot path (Test 1)
386 overlaid over Google Earth view with three pinpointed positions, whereas
387 the corresponding multi-view output is displayed in Figure 13.

388 Point clouds are collected and assembled in the w frame using estimates
389 of both absolute position and orientation of the sensor box. In Figure 14, the
390 EIF observer output is used to merge point clouds collected by the frontal
391 camera. Figure 15(a) shows, instead, about 20 m of merged point clouds
392 from all cameras using 6DoF odometry provided by the EIF. This map can be
393 processed to extract high-level information about the crop, such as vegetation
394 indexes and morphological information. As an example, Figures 15(b) and
395 (c) show the map of Figure 15 (a) augmented with Green-Red Vegetation
396 Index (GRVI) and crop elevation information, respectively.

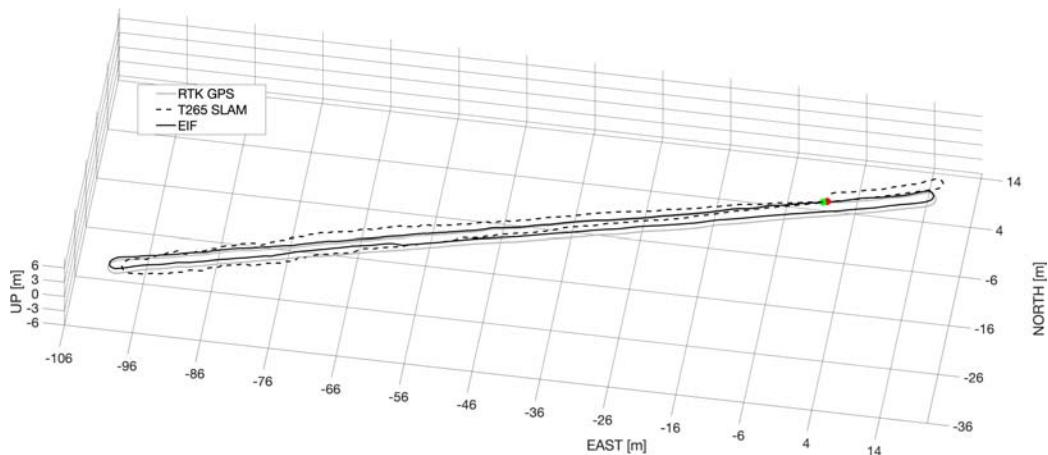


Figure 9: Localization results for a second run along the same path of Figure 8 (Test 2) from RTK-GPS only (solid grey line), T265 only (dashed black line) and EIF (solid black line). Start and stop positions for EIF trajectory are denoted by green and red dot, respectively.

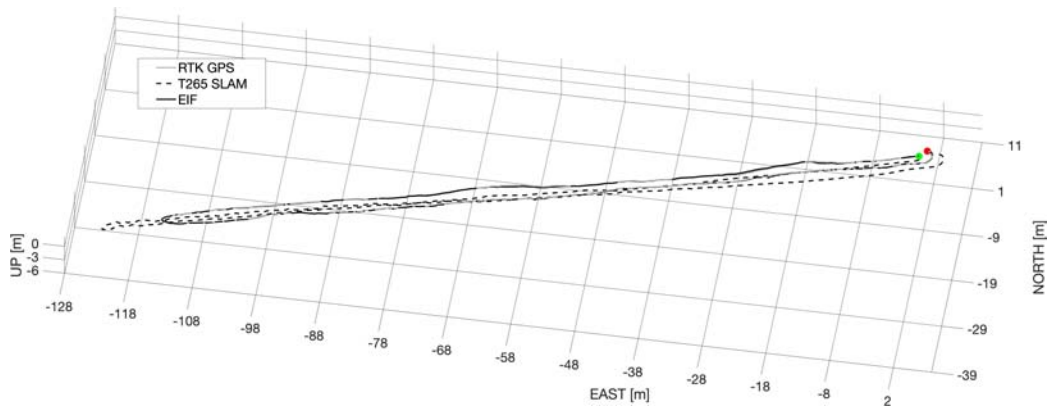


Figure 10: Localization results for Test 3 from RTK-GPS only (solid grey line), T265 only (dashed black line) and EIF (solid black line). Start and stop positions for EIF trajectory are denoted by green and red dot, respectively. In this test, the T265 estimate is substantially degraded and the EIF mainly relies on GPS.

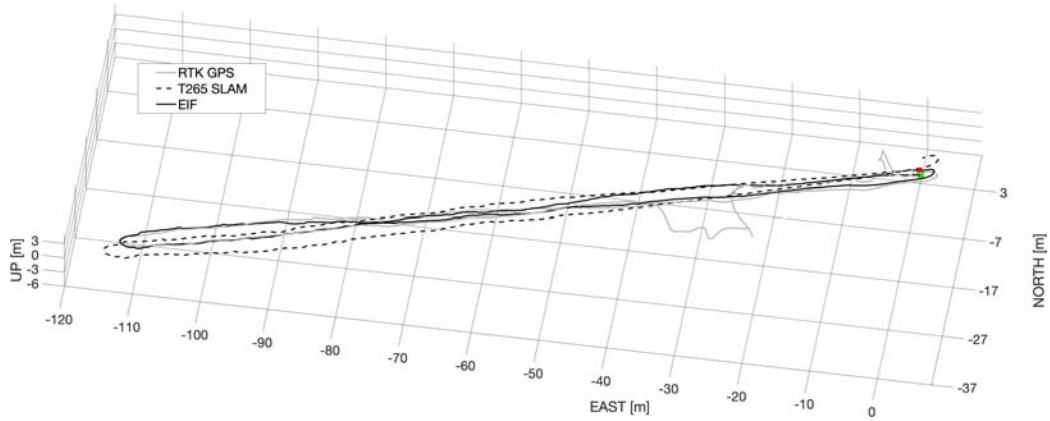


Figure 11: Localization results for Test 4 from RTK-GPS only (solid grey line), T265 only (dashed black line) and EIF (solid black line). Start and stop positions for EIF trajectory are denoted by green and red dot, respectively. In this test, EIF is able to compensate poor GPS signal quality based on T265 information.

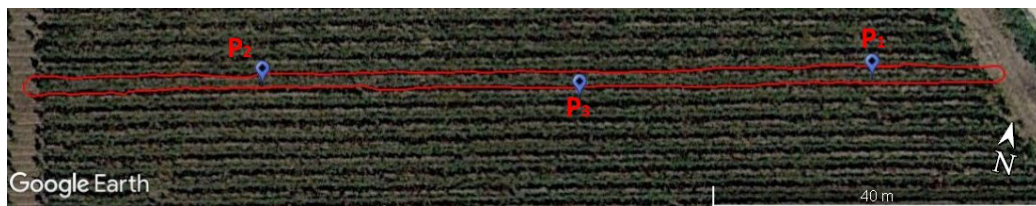


Figure 12: EIF-derived path overlaid on Google Earth view for Test 1. Three successive positions of the robot are pinpointed. For these positions, the corresponding visual data are shown in Figure 13.

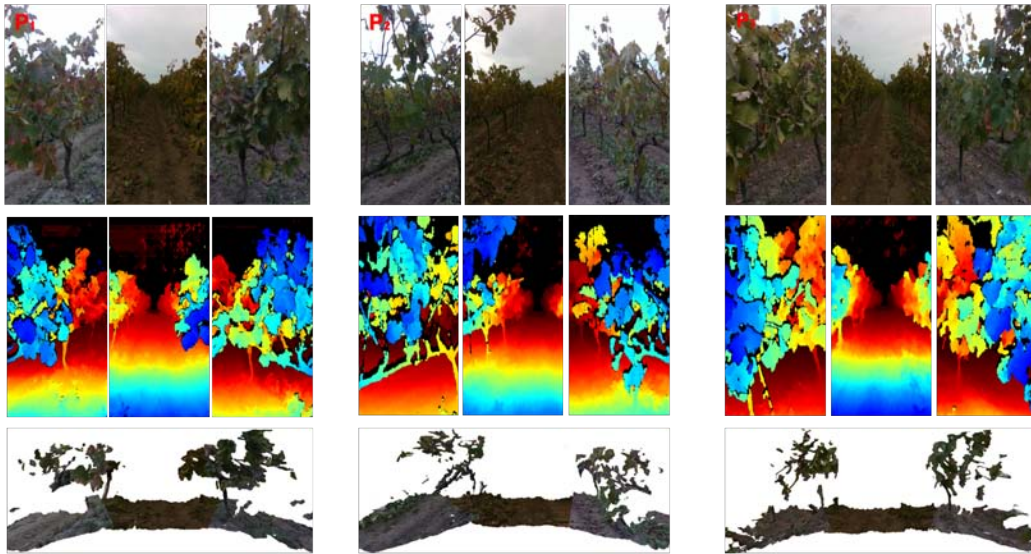


Figure 13: Output of the multi-view camera system for three robot locations along the path (Test 1): (first row) color images, (second row) depth images obtained from IR stereo reconstruction and (third row) multi-view 3D point cloud.

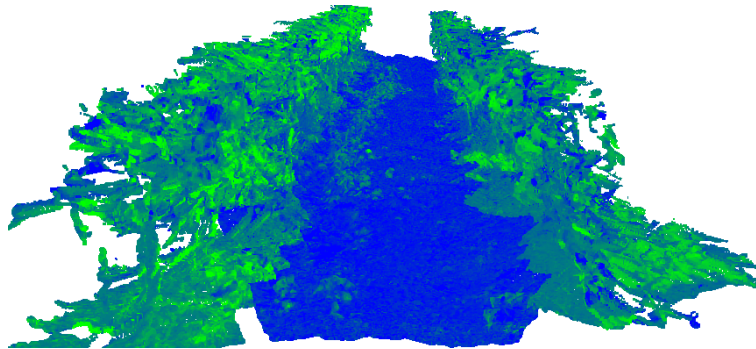


Figure 14: Mapping results (Test 1): upper view of the terrain map reconstructed by the central camera. The robot trajectory estimated by the sensor fusion approach is also overlaid.

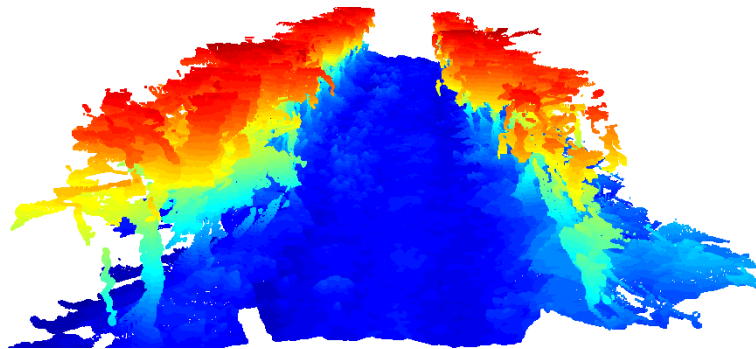
397 Accurate localization information is essential to assemble subsequent point
 398 clouds acquired by the D435 cameras and build the environment map. This
 399 can be clearly seen in Figure 16, where two different 6DoF localization sources
 400 are compared. In detail, Figure 16(a) shows a group of point clouds badly
 401 assembled with synced data of GPS for position and IMU for orientation
 402 when RTK correction are missing. Figure 16(b) is obtained using the EIF
 403 for the same time span, clearly showing the improvement in point cloud
 404 assembling.



(a)

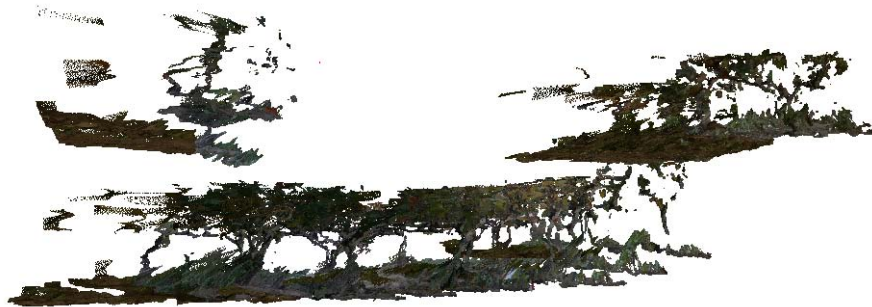


(b)



(c)

Figure 15: Closeup of the multi-view map for Test 1 (first 20 m): (a) RGB, (b) GRVI and (c) elevation map. In (b), green points refer to vegetation, whereas blue points correspond to non-vegetated parts. Lighter green denotes higher GRVI values. In (c), a jet colormap is used to represent point height with respect to ground.



(a)



(b)

Figure 16: Mapping results (Test 1): closeup of loop closure before (a) and after (b) EIF correction.

405 7. Conclusions

406 In this paper, the development, implementation and testing of a multi-
407 view RGB-D sensing device is presented. The system is intended to be
408 mounted on an agricultural ground robot for in-field proximal monitoring
409 of high-value crops. A multi-view mapping approach to combine information
410 from multiple visual and localization sensors and produce a high-resolution
411 3D reconstruction of agricultural environments is described. It is based on an
412 EIF algorithm to fuse information from RTK-GPS, IMU and visual-inertial
413 SLAM to obtain an accurate estimation of the vehicle position in the field.
414 Then, on the basis of localization data, subsequent point clouds reconstructed
415 by the RGB-D sensors during robot motion can be assembled to generate a
416 high-resolution map of the surveyed environment. Results of dedicated tests
417 performed in a commercial vineyard are presented, showing the effectiveness
418 of the proposed system for in-field data gathering in an automatic and non-
419 invasive way.

420 Future work will include the processing of the maps using supervised or un-
421 supervised classification methods to generate semantic representations of the

422 environment, which can be used to improve vehicle autonomy and safety. Re-
423 search will focus on the integration of output maps into Farm Management
424 Information Systems (FMIS) to enable map-based control of agricultural ap-
425 plications and machinery. In this respect, future efforts will be devoted to
426 address the real-time challenge by using the multi-view maps for online nav-
427 igation of autonomous agricultural vehicles. Furthermore, methods for iden-
428 tification and mapping of anomalies, such as weeds, as well as the extraction
429 of geometric measurements, such as plant volume/height estimates, will be
430 integrated to enable precision farming practices. This would also improve
431 the cost-benefit ratio of the sensor suite.

432 Acknowledgments



433 This work was funded by the following research programs: European
434 Union’s Horizon 2020 research and innovation programme under grant agree-
435 ment “ATLAS - Agricultural Interoperability and Analysis System” (No.
436 857125); ERA-NET ICT-AGRI-FOOD COFUND ”ANTONIO-multimodal
437 sensing for individual pLANT phenOtyping in agriculture robotics” (No.
438 41946); “E-crops - Technologies for Digital and Sustainable Agriculture”
439 funded by the Italian Ministry of University and Research (MUR) under
440 the PON Agrifood Program (No. ARS01_01136); CNR DIITET project
441 “DIT.AD022.180 Transizione industriale e resilienza delle Società post-Covid19
442 (FOE 2020)”, sub task activity “Agro-Sensing”. The authors are grateful to
443 Cantina San Donaci agricultural farm for hosting experimental tests. The
444 administrative and technical support by Michele Attolico and Giuseppe Bono
445 is also gratefully acknowledged.

446 *Disclaimer:* The sole responsibility for the content of this publication
447 lies with the authors. It does not necessarily represent the opinion of the
448 European Union. Neither the EASME nor the European Commission is re-
449 sponsible for any use that may be made of the information contained therein.

450 Author contributions

451 **Fabio Vulpi:** Conceptualization, Methodology, Software, Validation,
452 Writing-Original draft; **Roberto Marani:** Software, Writing-Original draft;

453 **Antonio Petitti**: Software, Writing-Original draft; **Giulio Reina** and
454 **Annalisa Milella**: Conceptualization, Methodology, Software, Validation,
455 Writing-Original draft, Writing- Reviewing and Editing, Supervision.
456

457 Appendix

Quaternion representations are convenient for composition of rotations and coordinate transformations. The unit quaternion $\mathbf{q} = [q_x, q_y, q_z, q_w]$, is uniquely mapped to a rotation matrix \mathbf{R} and describes the transformation between two reference frames as a rotation of a certain angle θ around the direction vector \vec{n} following Equation A.1.

$$\mathbf{q} = [n_x \sin \frac{\theta}{2}, n_y \sin \frac{\theta}{2}, n_z \sin \frac{\theta}{2}, \cos \frac{\theta}{2}] \quad (\text{A.1})$$

Denoting with s_1 and \vec{v}_1 respectively the scalar and vector part of quaternion $\mathbf{q}_1 = [\vec{v}_1, s_1] = [v_{1x}, v_{1y}, v_{1z}, s_1]$, and with s_2, \vec{v}_2 the scalar and vector part of quaternion \mathbf{q}_2 , the operation of quaternion product can be expressed as

$$\mathbf{q}_2 \mathbf{q}_1 = [s_2 \vec{v}_1 + s_1 \vec{v}_2 + \vec{v}_2 \times \vec{v}_1, s_2 s_1 - \vec{v}_2 \cdot \vec{v}_1] \quad (\text{A.2})$$

where quaternion product symbol has been omitted for readability, whereas $\vec{v}_2 \cdot \vec{v}_1$ denotes dot product between vectors \vec{v}_1 and \vec{v}_2 and finally $\vec{v}_2 \times \vec{v}_1$ denotes cross product between the two vectors. Quaternion product is a non-commutative operation and returns a quaternion that represents the orientation obtained after the sequence of transformations \mathbf{q}_1 and then \mathbf{q}_2 . The norm of a quaternion \mathbf{q} is denoted as $\|\mathbf{q}\|^2 = q_x^2 + q_y^2 + q_z^2 + q_w^2$. The conjugate of quaternion $\mathbf{q} = [\vec{v}, s]$ is represented as $\mathbf{q}^* = [-\vec{v}, s]$, while its inverse $\mathbf{q}^{-1} = \frac{\mathbf{q}^*}{\sqrt{\|\mathbf{q}\|^2}}$. For a unit quaternion we have $\|\mathbf{q}\|^2 = 1$ so its conjugate coincides with its inverse. All quaternions describing orientation in 3-D space are unit quaternions. The normalized quaternion denoted as $\|\mathbf{q}\| = \frac{\mathbf{q}}{\sqrt{\|\mathbf{q}\|^2}}$ has a unit norm and each of its components are divided by $\sqrt{\|\mathbf{q}\|^2}$. Let us denote with \mathbf{q}_{B-A} the quaternion describing orientation of frame A with respect to frame B written in frame B , its inverse is $\mathbf{q}_{B-A}^{-1} = \mathbf{q}_{A-B}$. Then, composition of rotations can be obtained in a convenient form as

$$\mathbf{q}_{C-A} = \mathbf{q}_{C-B} \mathbf{q}_{B-A} \quad (\text{A.3})$$

Consider the position vector \vec{p}_A in frame A , then its projection in reference frame B can be obtained as

$$[\vec{p}_B, 0] = \mathbf{q}_{B-A}[\vec{p}_A, 0]\mathbf{q}_{A-B} \quad (\text{A.4})$$

Finally, denoting with $\vec{\omega}_B(t)$ the angular velocity of moving frame B in its reference frame A , the derivative of the quaternion $\mathbf{q}_{A-B}(t)$ expressed in the inertial frame A can be computed as

$$\frac{d\mathbf{q}_{A-B}(t)}{dt} = \frac{1}{2}\mathbf{q}_{A-B}(t)[\vec{\omega}_B(t), 0] \quad (\text{A.5})$$

458 References

- 459 [1] G. Reina, A. Milella, R. Rouveure, M. Nielsen, R. Worst,
 460 M. R. Blas, Ambient awareness for agricultural robotic ve-
 461 hicles, *Biosystems Engineering* 146 (2016) 114–132, spe-
 462 cial Issue: Advances in Robotic Agriculture for Crops.
 463 doi:<https://doi.org/10.1016/j.biosystemseng.2015.12.010>.
- 464 [2] Gazebo simulator, <http://gazebosim.org/>, accessed: 2021-12-06.
- 465 [3] A. Matese, P. Toscano, S. F. Di Gennaro, L. Genesio, F. P. Vaccari,
 466 J. Primicerio, C. Belli, A. Zaldei, R. Bianconi, B. Gioli, Intercomparison
 467 of UAV, aircraft and satellite remote sensing platforms for precision
 468 viticulture, *Remote Sensing* 7 (3) (2015) 2971–2990.
- 469 [4] L. Comba, A. Biglia, D. Ricauda Aimonino, P. Gay, Unsupervised de-
 470 tection of vineyards by 3d point-cloud uav photogrammetry for precision
 471 agriculture, *Computers and Electronics in Agriculture* 155 (2018) 84–95.
- 472 [5] A. Gongal, S. Amatya, M. Karkee, Q. Zhang, K. Lewis, Sen-
 473 sors and systems for fruit detection and localization: A re-
 474 view, *Computers and Electronics in Agriculture* 116 (2015) 8–19.
 475 doi:<https://doi.org/10.1016/j.compag.2015.05.021>.
- 476 [6] Z. Li, R. Guo, M. Li, Y. Chen, G. Li, A review of
 477 computer vision technologies for plant phenotyping, *Com-
 478 puters and Electronics in Agriculture* 176 (2020) 105672.
 479 doi:<https://doi.org/10.1016/j.compag.2020.105672>.

- 480 [7] J. Ma, K. Du, L. Zhang, F. Zheng, J. Chu, Z. yu Sun, A segmentation
481 method for greenhouse vegetable foliar disease spots images using color
482 information and region growing, *Comput. Electron. Agric.* 142 (2017)
483 110–117.
- 484 [8] P. Jiang, Y. Chen, B. Liu, D. He, C. Liang, Real-time detection of
485 apple leaf diseases using deep learning approach based on improved
486 convolutional neural networks, *IEEE Access* 7 (2019) 59069–59080.
487 doi:10.1109/ACCESS.2019.2914929.
- 488 [9] J. Mack, C. Lenz, J. Teutrine, V. Steinhage, High-precision 3d detec-
489 tion and reconstruction of grapes from laser range data for efficient phe-
490 notyping based on supervised learning, *Computers and Electronics in*
491 *Agriculture* 135 (2017) 300–311.
- 492 [10] A. Milella, R. Marani, A. Petitti, G. Reina, In-field high throughput
493 grapevine phenotyping with a consumer-grade depth camera, *Comput-*
494 *ers and Electronics in Agriculture* 156 (2019) 293–306.
- 495 [11] A. Milella, G. Reina, M. Nielsen, A multi-sensor robotic platform for
496 ground mapping and estimation beyond the visible spectrum, *Precision*
497 *Agric* 20 (2019).
- 498 [12] A. Milella, G. Reina, 3D reconstruction and classification of natural en-
499 vironments by an autonomous vehicle using multi-baseline stereo, *Intel*
500 *Serv Robotics* 7 (2014) 79–92.
- 501 [13] Y. Chéné, D. Rousseau, P. Lucidarme, J. Bertheloot, V. Caffier,
502 P. Morel, É. Belin, F. Chapeau-Blondeau, On the use of depth cam-
503 era for 3D phenotyping of entire plants, *Computers and Electronics in*
504 *Agriculture* 82 (2012) 122–127.
- 505 [14] R. P. Devanna, A. Milella, R. Marani, S. P. Garofalo, G. A. Vivaldi,
506 S. Pascuzzi, R. Galati, G. Reina, In-field automatic identification of
507 pomegranates using a farmer robot, *Sensors* 22 (15) (2022).
- 508 [15] I. C. Condotta, T. M. Brown-Brandl, S. K. Pitla, J. P. Stinn, K. O.
509 Silva-Miranda, Evaluation of low-cost depth cameras for agricultural ap-
510 plications, *Computers and Electronics in Agriculture* 173 (2020) 105394.

- 511 [16] M. Imperoli, C. Potena, D. Nardi, G. Grisetti, A. Pretto,
512 An effective multi-cue positioning system for agricultural robotics,
513 IEEE Robotics and Automation Letters 3 (4) (2018) 3685–3692.
514 doi:10.1109/LRA.2018.2855052.
- 515 [17] N. Habibie, A. M. Nugraha, A. Z. Anshori, M. A. Ma’sum, W. Jat-
516 miko, Fruit mapping mobile robot on simulated agricultural area in
517 Gazebo simulator using simultaneous localization and mapping (SLAM),
518 in: 2017 International Symposium on Micro-NanoMechatronics and Hu-
519 man Science (MHS), 2017, pp. 1–7. doi:10.1109/MHS.2017.8305235.
- 520 [18] J. A. Hage, S. Mafrica, M. E. B. E. Najjar, F. Ruffier, Informational
521 framework for minimalistic visual odometry on outdoor robot, IEEE
522 Transactions on Instrumentation and Measurement 68 (8) (2019) 2988–
523 2995.
- 524 [19] W. Xu, Y. Cai, D. He, J. Lin, F. Zhang, Fast-lio2: Fast direct lidar-
525 inertial odometry, IEEE Transactions on Robotics (2022).
- 526 [20] I. Ullah, X. Su, J. Zhu, X. Zhang, D. Choi, Z. Hou, Evaluation of local-
527 ization by extended Kalman filter, unscented Kalman filter, and particle
528 filter-based techniques, Wireless Communications and Mobile Comput-
529 ing 2020 (2020) 1–15.
- 530 [21] G. Reina, A. Leanza, G. Mantriota, Model-based observers for vehicle
531 dynamics and tyre force prediction, Vehicle System Dynamics (2021)
532 1–26.
- 533 [22] A. Petitti, F. Vulpi, R. Marani, A. Milella, A self-calibration approach
534 for multi-view RGB-D sensing, in: E. Stella (Ed.), Multimodal Sensing
535 and Artificial Intelligence: Technologies and Applications II, Vol. 11785,
536 International Society for Optics and Photonics, SPIE, 2021, pp. 50 – 55.
537 doi:10.1117/12.2595165.
- 538 [23] H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard,
539 L. Kavraki, S. Thrun, Principles of Robot Motion, The MIT Press,
540 Cambridge, MA, USA, 2004.
- 541 [24] M. Corsini, P. Cignoni, R. Scopigno, Efficient and flexible sam-
542 pling with blue noise properties of triangular meshes, IEEE Transac-

- 543 tions on Visualization and Computer Graphics 18 (6) (2012) 914–924.
544 doi:10.1109/TVCG.2012.34.
- 545 [25] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, G. Taubin,
546 The ball-pivoting algorithm for surface reconstruction, IEEE Transac-
547 tions on Visualization and Computer Graphics 5 (4) (1999) 349–359.
548 doi:10.1109/2945.817351.