

# Development of an RFID Inventory Robot (AdvanRobot)

Marc Morenza-Cinos<sup>1</sup>, Victor Casamayor-Pujol<sup>1</sup>, Jordi Soler Busquets<sup>1</sup>, José Luis Sanz<sup>2</sup>, Roberto Guzmán<sup>3</sup>, and Rafael Pous<sup>1</sup>

<sup>1</sup> Universitat Pompeu Fabra,  
Roc Boronat 138, 08018 Barcelona, Spain  
[www.upf.edu](http://www.upf.edu)

<sup>2</sup> Keonn Technologies S.L.,  
Pere IV 78-84, 08005 Barcelona, Spain  
[www.keonn.com](http://www.keonn.com)

<sup>3</sup> Robotnik Automation S.L.L.,  
Ciudad de Barcelona 3-A, 46988 Valencia, Spain  
[www.robotnik.eu](http://www.robotnik.eu)

**Abstract.** AdvanRobot proposes a new robot for inventorying and locating all the products inside a retail store without the need of installing any fixed infrastructure. The patent pending robot combines a laser-guided autonomous robotic base with a Radio Frequency Identification (RFID) payload composed of several RFID readers and antennas, as well as a 3D camera.

AdvanRobot is able not only to replace human operators, but to dramatically increase the efficiency and accuracy in providing inventory, while also adding the capacity to produce store maps and product location. Some important benefit of the inventory capabilities of AdvanRobot are the reduction in stock-outs, which can cause a drop in sales and are the most important source of frustration for customers; the reduction of the number of items per reference maximizing the number of references per square meter; and reducing the cost of capital due to overstocking [1, 7]. Another important economic benefit expected from the inventorying and location capabilities of the robot is the ability to efficiently prepare on-line orders from the closest store to the customer, allowing retailers to compete with the likes of Amazon (a.k.a. omnichannel retail). Additionally, the robot enables to: produce a 3D model of the store; detect misplaced items; and assist customers and staff in finding products (wayfinding).

**Keywords:** Professional service robotics applications with ROS, Logistic robots, Service robotics, Autonomous robots, Mobile robots, RB1-Base, RFID

# Table of Contents

Table of Contents .....	1
<i>Marc Morenza-Cinos, Victor Casamayor-Pujol, Jordi Soler Busquets (Universitat Pompeu Fabra), José Luis Sanz (Keonn Technologies), Roberto Guzmán (Robotnik Automation), Rafael Pous (Universitat Pompeu Fabra)</i>	
1 Introduction .....	4
2 Background .....	6
2.1 RFID technology .....	6
2.2 Inventory systems .....	7
3 AdvanRobot Overview .....	8
3.1 Design .....	8
3.1.1 Robotic Base. ....	9
3.1.2 RFID Payload. ....	9
3.1.3 Interconnection. ....	10
3.1.4 Comparison with other RFID Robots. ....	11
3.2 Architecture .....	11
3.2.1 Task Manager. ....	12
3.3 Navigation .....	13
3.3.1 Recognition Stage. ....	13
3.3.2 Inventory Stage. ....	14
3.3.3 Integration. ....	15
3.3.4 Challenges. ....	16
3.4 Human-Robot Interaction .....	17
3.4.1 Launching the <i>Recognition Stage</i> . ....	18
3.4.2 Launching an <i>Inventory Mission</i> . ....	19
3.4.3 Human-Robot Interface. ....	19
3.4.4 Challenges. ....	20
4 Simulation .....	21
5 Experimental Results .....	21
5.1 Navigation Validation .....	21
5.2 RFID Identification Accuracy .....	23
5.3 Operation by Store Associates .....	24
6 Ongoing Developments .....	25
6.1 Exploration for 3D Mapping .....	25
6.1.1 ROS Package: Exploration Framework. ....	26
6.1.2 ROS API. ....	28
6.2 Location of RFID items .....	29
7 Future Work .....	29
7.1 Collaborative Inventorying .....	29
7.2 UAVs and AdvanRobot Collaborative System .....	30
7.3 Applications derived from product location .....	31

Development of an RFID Inventory Robot (AdvanRobot)	3
7.4 Simulation.....	31

## 1 Introduction

In this chapter a solution for smart retail that combines robotics and Radio Frequency IDentification (RFID) technology is presented.

Traditional retail (a.k.a. "brick-and-mortar" retail) is facing fierce competition from on-line retail. While traditional retail still keeps some advantages (e.g. physical contact with the products or immediate fulfillment), on-line retail continues to offer more and more advantages that are increasingly appealing to customers (e.g. easy to find products, no stock outs, in depth information, recommendations, user opinions, social networks integration, etc.) Also, on-line retail has at its disposal a wealth of data about its customers' clickstream that can be leveraged through sophisticated analysis to offer personalized and targeted websites against which a generic "one model fits all" physical stores cannot compete. As a result, on-line retail is growing with double digits, while many retailers continue to close physical stores.

RFID offers an opportunity for traditional retailers to fight back. If every product in the retail store is tagged with RFID, it is given an Electronic Product Code (EPC), which is universally unique for each item. By placing RFID equipment at the store every relevant event of the product can be detected. Many leading retailers such as Kohl's, Decathlon, Inditex or Marks & Spencers have already deployed RFID technology in their stores.

The most obvious, and most common, application of RFID in the store is for inventory. Most commonly, RFID-based inventories are done by using handheld RFID readers, that store associates use to scan every shelf, rack and fixture in the store. A typical fashion store in a typical shopping mall of about 1.000  $m^2$ , and about 10.000 items can be completely inventoried by a single associate in under 60 minutes. The same process using barcode technology would typically require a team of 3 to 5 persons working for one or two full days (double or triple counts are typically necessary in this case to reach an acceptable accuracy).

However, although the theoretical accuracy of RFID inventory using handheld readers is above 99%, retailers using this method report actual accuracies of between 80 and 90%. The difference lies in human errors. Inventory taking is a very tedious process, and the staff doing it frequently forget a shelf, an aisle, or an entire section. The layout of a retail store is typically not regular, and it is very easy, especially in larger stores, for associates to get confused and believe that they have already scanned a part of the store when they in fact have not. Also, the repetitive movements involved in scanning the store have been linked to injuries, raising an issue of health in the work place.

Whenever humans are faced with tedious repetitive physical tasks, robots are the ideal candidates to overtake them, especially when there are health risks involved. Keonn Technologies, a manufacturer of RFID solutions for the retail sector, had the idea to combine a standard robotic base with an RFID payload composed of standard components to create a robotic inventory system for the retail store. The idea included some important insights on how to couple the navigation and the RFID systems to increase efficiency and the accuracy of the inventory process. In 2013 Keonn filed a patent and presented the first commer-

cial prototype of an RFID inventory robot at the RFID Journal Live 2013 show, where it was selected as the winner of the "Coolest demo award" [15]. Since 2013 Keonn has taken the latest versions of AdvanRobot to the same show, where it has raised a lot of interest among retailers and in the RFID industry in general. During this time, Keonn, with the collaboration of the Ubiquitous Computing Applications Lab (UbiCA Lab) at Pompeu Fabra University, and the robotics company Robotnik has continuously improved the product, tested it extensively in large retail stores around the world, and established agreements with some of the most important players in the RFID for retail industry. AdvanRobot is the only robotic system for inventory designed by a multidisciplinary team of RFID specialists, robotics specialists, and academia. The resulting product, AdvanRobot, is now a part of Keonn's portfolio of RFID solutions for retail.

AdvanRobot is able to inventory a very large store during the 10 to 12 hours in which a store is normally closed. For the same job, at least 4 associates with RFID handheld readers would be required, making the Return on Investment (ROI) very high. Additionally, AdvanRobot never forgets a shelf, an aisle or a section, and the measured accuracy has always been above 99.5% of all tagged products in the store. In fact, AdvanRobot is the most accurate instrument to perform an inventory.

In addition, AdvanRobot is able to provide not only inventory but also the location of the products on the store layout and a 3D model of the store. This is considered of high value for retailers to detect misplaced items, to help customers find products and associates fulfill on-line orders.

In this chapter, the following topics are covered:

- First, we present a background section on RFID technology and its applications to retail.
- Second, AdvanRobot's overview is given including its design and architecture, analyzing the specific navigation strategies for inventorying a store and finishing with the human-robot interaction.
- Third, a short introduction to AdvanRobot simulation is provided.
- Fourth, we describe the results of the tests carried out in actual retail environments.
- Fifth, ongoing developments are explained. We introduce a framework for the exploration and mapping of 3D environments. The ROS package is named *cam\_exploration* and the code is publicly available at UbiCALab's github account<sup>4</sup>.
- Sixth, we discuss the developments considered for future versions of AdvanRobot.

Due to the nature of the project some of the core packages of AdvanRobot are not publicly available. Nonetheless, a set of packages for a basic simulation are available<sup>5</sup>.

<sup>4</sup> [https://github.com/UbiCALab/cam\\_exploration](https://github.com/UbiCALab/cam_exploration)

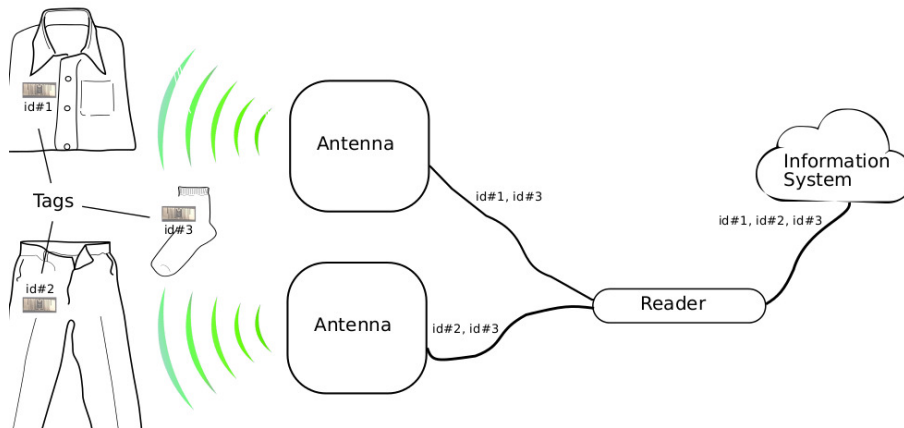
<sup>5</sup> <https://github.com/UbiCALab/advanrobot>

A video<sup>6</sup> of AdvanRobot in a store introduces its operation and main features.

## 2 Background

### 2.1 RFID technology

The central component of RFID technology is the RFID tag, composed of a chip mounted on a low cost antenna (usually made from etched aluminium). When an RFID reader sends an interrogating wave, all tags within the reach of the reader's antenna respond with a unique code, which the reader communicates through a wired or wireless network to an information system that makes use of this code for a given application. Figure 1 illustrates the different components of a typical RFID system.



**Fig. 1.** Components of a typical RFID system. The Reader interrogates the environment by sending an RF signal through the Antennas. Tags, attached to products, reply with their unique identifier. The Reader communicates the data to an Information System for its exploitation.

In most cases, RFID tags are passive, and obtain the energy to respond by rectifying the interrogating signal wave. In some cases, besides a code, the RFID tag has a limited amount of user memory.

RFID technology has been around for decades now. However, it was not until 1999 when the AutoID Center at MIT redefined the frequency, the protocols and the code standards [16], that RFID started to become a widely adopted technology, especially in the retail sector. These standards were later acquired and

<sup>6</sup> <https://youtu.be/V72Ep4s9T4o>

are now managed by GS1 <sup>7</sup>, the global organization also managing commercial barcode standards.

As opposed to previous standards that use the low and high frequency bands (LF and HF), the new standard uses the ultra high frequency band (UHF) [6]. The band in Europe (ETSI standard) is from 865.6 to 867.6 MHz, and in the USA (FCC standard) from 902 to 920 MHz.

An RFID reader in the UHF band can read tags at a distance of up to 10 meters as opposed to less than 2m in the LF and HF bands. RFID antennas have beam widths normally between 30 and 90 degrees. In a typical scenario the reader can identify hundreds, sometimes thousands of tags simultaneously, for which the Gen2 protocol [4] incorporates anticollision protocols allowing read rates of hundreds of tags per second.

The weight and battery constraints of handheld RFID readers limit the maximum power and in consequence the read range to between one and two meters. A robotic system, on the other hand carries a high capacity battery, which can operate several readers at full power, each connected to several antennas, each of them with a read range of between 2 to 5 meters. In consequence, a robotic RFID inventory system can have the equivalent reading capability of 10 to 20 handheld readers.

## 2.2 Inventory systems

Due to errors, theft, misplacements and other reasons, actual inventories diverge significantly from theoretical inventories in the shop floor, typically by 10 to 20% [8]. Most retailers will use barcodes for inventorying, but inventories based on barcodes are expensive, disruptive, can only be done every few months, and their accuracy is typically no higher than 95%. This situation results in frequent stock outs, frustrated customers, expensive preventive overstocking, and in the impossibility to source on-line orders directly from the stores, which would allow retailers to effectively compete with online retailers [14]. In contrast, RFID-based inventories are much more affordable, non disruptive, and the accuracy is usually above 99%.

There are several options to inventory a store based on RFID technology. First, handheld RFID devices may be used to accurately take inventory of objects tagged with RFID tags. Second, ceiling mounted readers with fixed or steerable beam antennas can be used to inventory RFID-tagged objects. Third, smart shelves or fixtures, incorporating RFID antennas and readers can be used to continuously inventory the objects they contain, as long as they are tagged with RFID. And fourth, autonomous robots [13] or UAVs [17] can be used to inventory all objects in a space, also using RFID tagging.

Handheld readers cannot, by themselves, provide any information about the location of the objects within the space being inventoried, while the other three methods can provide location with different degrees of accuracy. The first and fourth methods can provide frequent but non continuous inventory, while the

<sup>7</sup> <http://www.gs1.org/>

second and third can provide quasi real time inventory (a.k.a. "near" time inventory). The first and fourth methods do not require any fixed infrastructure installation, and the second and third methods do. The four methods also differ in the cost of hardware and the cost of labor they require. Table 1 summarizes the above comparison.

**Table 1.** Comparison of RFID inventory methods

Method	Location accuracy	Inventory frequency	Fixed infrastructure	Hardware cost	Labor cost
Handheld reader	No location	Every few days or weeks	No	Very low	High
Ceiling mounted readers	2 m	Every few minutes	Yes	High	None
Smart shelves and fixtures	50 cm	Every few seconds	Yes	Very high	None
Autonomous robot	1-2 m	Every day	No	Low	Very low

### 3 AdvanRobot Overview

AdvanRobot is an autonomous mobile robot that takes inventory of RFID labeled products in large retail stores. Therefore, by using AdvanRobot, taking inventory becomes an automatized task. In addition, complementary features revealed after its initial concept, for instance the location of the RFID labeled products and the generation of 3D maps of the environment. This section is developed as follows: first, the AdvanRobot is described detailing its design and characteristics; second, a high-level overview of the system architecture is given; third, the navigation strategies for inventorying a store are defined; and finally, the human-robot interaction is detailed.

#### 3.1 Design

AdvanRobot is designed in two main systems: the robotic base and the payload. Briefly, the robotic base is a ROS based autonomous mobile robotic base that is in charge of satisfying all the requirements that the payload needs for inventorying. It provides power, a safe and reliable navigation, and connectivity with the environment. The payload is the system in charge of performing the main task of the robot which is taking inventory.



In addition, it incorporates a web interface for the human robot interaction that can run in any web browser. This interface allows the interaction with the user from a very high-level simplifying all tasks and ensuring that everything runs as required. The Human-robot interface is detailed in Sect. 3.4.3.

**3.1.1 Robotic Base.** The robotic base is the model RB-1 manufactured by Robotnik<sup>8</sup>. It is a circular base with differential wheels allowing an excellent maneuverability in narrow aisles since its turning radius is 0. Moreover, it has a load capacity of 50 kg and provides high stability and damping. However, note that the RB-1 base used includes ad-hoc modifications. The base consists of three subsystems: the traction subsystem; the brain; and the power subsystem.

The traction subsystem includes two motorized and encoded wheels powered by servomotors, three omni-directional wheels, and two dampers for stability and overcoming floor irregularities. Due to its traction configuration it has differential drive capabilities. In addition, it has an emergency push (e-stop) button that cuts the servomotors power and immediately stops the robot.

Secondly, the brain, composed by the computer and electronics subsystem is in charge of controlling and connecting all the robot parts and providing all the intelligence required. Its main devices are the computer, a router that provides an access point for external connections and the sensors. The only embedded sensor is an IMU with two gyroscopes, an accelerometer and a compass. The RB-1 base also uses peripheral sensors: an optical (RGB) and depth (D) camera (RGBD camera) placed on top of the payload in order to detect obstacles, and a laser range finder. Finally it is also prepared for the installation of sonars in order to avoid those obstacles that can not be detected by the laser range finder or the RGBD camera such as mirrors, black surfaces and highly translucent materials which can be present in target environments. All the peripheral sensors connectors are easily accessible for their connection and disconnection.

Finally, the power subsystem consists of a lithium iron phosphate battery that provides more than 11 hours of autonomy. It also includes a battery management system (BMS) which controls the charging and discharging of the battery, and the electronics for recharging on a charging station.

**3.1.2 RFID Payload.** The RFID payload is the system in charge of taking inventory. It consists of three main parts:

First, AdvanRobot is equipped with 3 RFID readers<sup>9</sup>, which control 4 antennas each. However the robot can work with different configurations, using 1 to 3 readers combined with RFID multiplexers to control all the antennas.

Second, AdvanRobot mounts 6 RFID antennas per side, summing up a total of 12 RFID antennas<sup>10</sup>. The antennas are placed side by side in a way such that their reading areas overlap. In this fashion, there is a minimization of blind

<sup>8</sup> <http://wiki.ros.org/Robots/RB-1.BASE>

<sup>9</sup> <http://keonn.com/rfid-components/readers/advanreader-150.html>

<sup>10</sup> <http://keonn.com/rfid-components/antennas/advantenna-p22.html>

spots in what regards RFID readings. Hence, there is a degree of redundancy in the RFID subsystem. This is paramount in order to ensure a critical inventory accuracy. In the configuration shown in Fig. 2, AdvanRobot is equipped with the RFID antennas aforementioned. However, other types of antennas can be used in order to achieve different reading behaviors and scanning patterns.

Last, the structural subsystem, with the main aim of providing a physical support to the former: the RFID antennas and the RFID readers, in addition, the structural subsystem is foldable. AdvanRobot has been designed to read tags up to 2.75 m. As a result its height is slightly above 2 m. Therefore, the possibility of being folded implies the ability to traverse any door at the same time as being high enough to read all the products in a store.

Besides, the RGBD camera is linked to the top of the RFID payload. The reason for that is the maximization of the usable field of view of the camera. In what regards obstacle avoidance, this is crucial. For a safe navigation, AdvanRobot should detect any obstacle up to its height. This is better achieved observing the environment from the uppermost location of the payload.



**Fig. 2.** AdvanRobot in operation in a store. At the bottom, the lower circular part is the robotic base. On top of it, the RFID system, which is foldable. Finally, at the top of the RFID system the RGBD camera for obstacle detection.

**3.1.3 Interconnection.** Both parts are interconnected by two USB ports for the RGBD cameras; an RJ45 port to interface the RFID system; and a

connection for power supply. This connections are accessible through a small door in a side of the payload making the assembly and disassembly a very easy procedure.

**3.1.4 Comparison with other RFID Robots.** A comparative analysis is done based on the information available regarding other commercial RFID-equipped robots. At the moment, such information is not extensive. The comparison highlights the features that are explicitly different between the involved robots, which to our knowledge are:

- Tory<sup>11</sup> manufactured by *MetraLabs*
- StockBot<sup>12</sup> manufactured by *PAL Robotics*

From a structural point of view, both, AdvanRobot and Tory have a circular footprint of 25 cm radius, however, StockBot's footprint is not circular and its equivalent footprint radius is 35 cm. This directly impacts the minimum size of the aisles where the robot can navigate, limiting its versatility in some environments. Regarding height, AdvanRobot is above the others, but it can be folded.

With respect to the battery autonomy, Tory states providing 14h, Stockbot 8h and AdvanRobot 11h. In addition, Advanrobot is the one that recharges the battery in a shorter amount of time, it requires 2h while the StockBot needs 4h and Tory from 3 to 6h for a complete charge. The robots operational availability is defined as the ratio between the time the robot is working and the total robot time (working plus charging time). Therefore, AdvanRobot accounts for an operational availability of 84.6%, Tory of 70% and Stockbot of 66,6%. For instance, AdvanRobot would be operative for 84.6h over a complete period of 100h. The remaining 15.4h would be used for battery charging.

Focusing on the RFID system, StockBot has 8 integrated RFID antennas, Tory only mentions that it has integrated RFID antennas. AdvanRobot uses 12 antennas which characteristics can be selected among different options and from 1 to 3 readers upon application and user request. Thus, AdvanRobot provides an excellent versatility to adapt the RFID system to the environment.

Finally, to the best of our knowledge and from an operational point of view, AdvanRobot is the only one that is prepared to work by zones inside a shop floor. Such feature is explained in Sect. 3.4. Table 2 summarizes the analysis.

## 3.2 Architecture

The high-level architecture of the robot consists of 5 main blocks: User; Interface; Task manager; Navigation; and RFID. Figure 3 shows a schematic of the architecture. The Interface allows the communication between the user and the

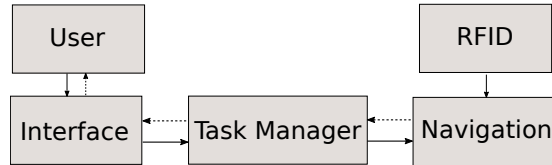
<sup>11</sup> <http://www.metralabs.com/en/shopping-rfid-robot>

<sup>12</sup> <http://pal-robotics.com/ca/products/stockbot>

**Table 2.** Summary of the comparative analysis of commercial inventory robots

	AdvanRobot	Tory	StockBot
Height [cm]	208	150	190
Equivalent footprint radius [cm]	25	25	35
Battery autonomy [h]	11	14	8
Charging time [h]	2	6	4
Operational Availability [%]	84.6	70.0	66.6
RFID system	1-3 readers 12 antennas	Not specified	2 readers 8 antennas

robot, it is the main component of the human-robot interaction. The *Task manager* basically translates the user requests into a set of actions. The Navigation block receives the actions and transforms them in commands for the movement of the robot. At the same time, the RFID system reads surrounding tags. The Navigation and RFID blocks interact in order to succeed in the selected task. The details of Navigation and RFID are explained in Sect. 3.3. Besides, the architecture follows the modularity of the robot’s design. The Navigation block corresponds to the autonomous base control and the RFID block to the RFID payload.



**Fig. 3.** System’s high-level architecture. Solid arrows indicate control while dashed arrows indicate control feedback. RFID plays a critical role in Navigation decisions in order to accomplish accuracy and time constraints.

**3.2.1 Task Manager.** The Task Manager is implemented in a node named *task\_manager*. It is the middleware that translates high-level user orders to lower-level control commands. It has been created in order to dispatch and monitor the tasks that the robot has to perform from a user perspective, meaning that it operates at a higher-level than the Navigation and RFID blocks.

*task\_manager* communicates with the interface via ROS Services<sup>13</sup>, in which the parameters of the service are the selected task and its options. Shortly, the node performs two main operations.

First it keeps the state for the robot, which is communicated to the user. Hence, the user knows the selected action status and progress. Additionally, the state prevents any interaction through the interface that could interfere with the current task. This state assignment allows the robot to work as a simplified finite state machine.

Second, it executes the selected task actions, using the ROS actionlib stack<sup>14</sup>. By means of actionlib, the node monitors the task, and if required it can also preempt the actions.

Also, *task\_manager* is subscribed to other nodes that publish the state of relevant parts of the robot. This has two main benefits. On one hand, ROS message passing facilitates monitoring all the relevant information, from the state of the RFID readers to the temperature of the motors. On the other hand, the *task\_manager* is the node that centralizes all the information and presents it to the user in a comprehensible way via the interface.

### 3.3 Navigation

AdvanRobot uses the ROS navigation stack<sup>15</sup> for a safe navigation in retail environments. AdvanRobot is configured to navigate in any wheelchair accessible retail space which are those comprised by aisles equal or greater than 70cm [5] [9]. It uses a range laser finder for simultaneous localization and mapping, widely known as SLAM [2] and an additional RGBD camera for obstacle detection in 3 dimensions. It is sonar-ready for the detection of mirrors and materials not reflective to light-spectrum. Before deploying AdvanRobot, an environment survey identifies potential risks for navigation and determines the need to use sonars, which can be plugged and played to clear the identified risks.

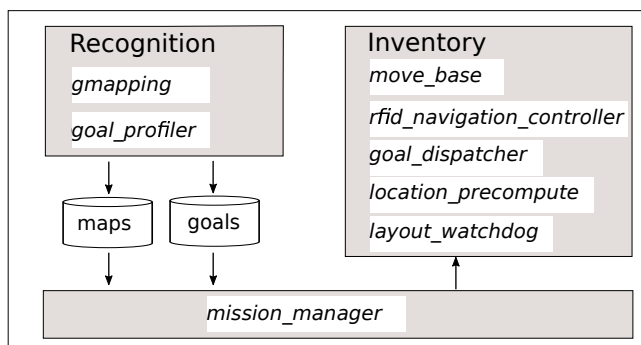
The navigation consists of a preparatory human assisted stage and a fully autonomous stage. The first is needed to get a baseline of the environment and it is called *Recognition stage*. During the *Recognition stage* AdvanRobot generates a map and records key spots for later navigation. Once the *Recognition stage* is completed successfully AdvanRobot is ready for the *Inventory stage* when the actual autonomous inventory taking is performed. In both stages RFID observations are used as inputs to support the optimal performance of AdvanRobot. Figure 4 shows schematically the Navigation parts explained next.

**3.3.1 Recognition Stage.** The aim of the recognition stage is providing a guided observation of the environment to AdvanRobot. By doing so, AdvanRobot learns the map of the zone intended for inventory and, by listening to the RFID readings, records the key spots where products are present. In practice,

<sup>13</sup> <http://wiki.ros.org/Services>

<sup>14</sup> <http://wiki.ros.org/actionlib>

<sup>15</sup> <http://wiki.ros.org/navigation>



**Fig. 4.** Building blocks of AdvanRobot’s navigation. In *Italic*, the names of the ROS nodes involved in each of the sub-blocks.

an operator brings AdvanRobot to a zone’s initial spot and using a remote control moves AdvanRobot close to the products to inventory. At the same time, a map is generated using ROS gmapping<sup>16</sup> and key spots are recorded by a purpose-developed ROS package: the *goal\_profiler*.

**3.3.2 Inventory Stage.** AdvanRobot performs inventory taking during the *Inventory stage*. Simultaneously, it does a pre-computation of RFID reads as a preparation for the latter offline location computation. The *Inventory stage* is triggered and controlled by the *mission\_manager* node.

Following the trigger of an inventory, the key spots recorded during the *Recognition stage* start to be dispatched in the form of navigation goals. In order to optimally dispatch navigation goals during the inventory stage, a ROS node, the *goal\_dispatcher*, monitors the progress of navigation and rearranges the goals online. For instance, if a key spot is visited unexpectedly due to path re-planning its corresponding navigation goals are cleared and not dispatched again.

Given a navigation goal, linear and angular velocities sent to the motors controller are commanded by the *rfid\_navigation\_controller* node, which monitors the progress of RFID reads in order to compute the following velocities. The *navigation\_control* node modulates the output of *move\_base*<sup>17</sup> in order to get the best inventory accuracy in the least time possible. For instance, if AdvanRobot moves at its maximum speed, which is 1m/s, in an environment with a high density of RFID labeled products, the RFID system has no time to identify all the products. Thus, the navigation needs an added control layer that takes into account the progress of RFID reads. Otherwise, inventory accuracy requirements are not met. Figure 5 shows the rqt diagram that relates the *navigation\_control* node to the *move\_base* node. In addition, the node *location\_precompute* matches RFID reads to the corresponding identification antenna pose using ROS trans-

<sup>16</sup> <http://wiki.ros.org/gmapping>

<sup>17</sup> [http://wiki.ros.org/move\\_base](http://wiki.ros.org/move_base)

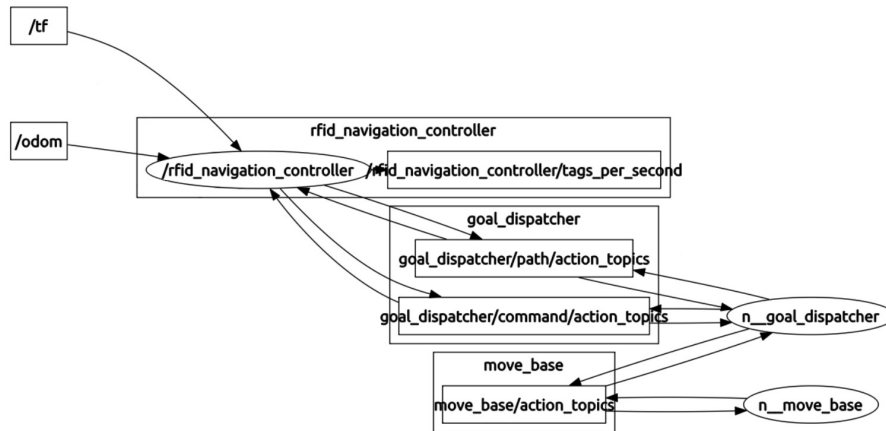


Fig. 5. Rqt diagram of the navigation controller node.

forms<sup>18</sup> lookups. Each RFID read is stored along with the antenna pose at the time of the identification for a posterior location computation. The computation of location is an ongoing development explained in Sect. 6.2

**3.3.3 Integration.** Linking properly the output of the *Recognition stage* to the subsequent *Inventory stages* is addressed by the *mission\_manager* node. A key system design feature is that AdvanRobot follows a divide and conquer strategy to conduct inventory missions. An important learning from field experience is that it is preferable to define zones of less than 1000 m<sup>2</sup> instead of working with a single larger zone. This is explained by three main reasons. The first, the ease of operation by the user in case an inventory of a specific zone or set of zones is needed. This has been validated with users during on-site pilots. The second, a less demanding computational cost. Working with big and precise maps implies a high computational cost. The third, a convenient modularity facing considerable layout changes and the consequent need of a re-recognition. Hence, AdvanRobot is designed to work by zones, following a divide and conquer strategy.

Given a set of zones, any combination is eligible and is defined as an *Inventory mission*. An *Inventory mission* comprises a set of consecutive *Inventory stages*. In this way, the user can select the zones to inventory as needed. Working with a set of zones requires a proper management of the *Recognition stage* output, which are maps and goals. For each zone, a pair map and set of goals is kept. Thus, an *Inventory mission* requires dispatching the proper map and goals in the appropriate order and timing to the navigation layer. The *mission\_manager* is the ROS node that performs the tasks of triggering and monitoring *Inventory stages* according to a defined *Inventory mission*.

<sup>18</sup> <http://wiki.ros.org/tf>

The divide and conquer strategy is implemented placing a zone identifier at the beginning of each defined zone. The zone identifier is the reference for AdvanRobot to know the actual zone. At the moment, the zone identifier is a QR code which is detected by the RGBD camera using the ROS package *ar\_track\_alvar*<sup>19</sup>. The automatic identification of zones enables the user to perform the recognition stage and afterwards launch an inventory mission without assistance. As well, zone identifiers are used by AdvanRobot to perform map transitions autonomously, commanded by the *mission\_manager*, since they define the relations between zones. Furthermore, QR codes can be used as a support to AdvanRobot’s location recovery and correction, which is discussed in Sect. 3.3.4.

An essential feature for a user is knowing when the *Recognition stage* needs to be rerun. By using ROS navigation stack properly, AdvanRobot is able to adapt to changes in the layout. However, if layout changes are significant, AdvanRobot may not be able to output a reliable localization and *Inventory missions* can fail. With the purpose of computing the need of rerunning a *Recognition stage* an algorithm is run by the node *layout\_watchdog*. The algorithm uses as inputs mainly but not only the success and time to reach goals and the reliability of the localization during the mission.

### 3.3.4 Challenges.

*Exploration.* The main challenge of the *Recognition stage* is suppressing the need of human assistance. For that, exploration has been considered and preliminary tests conducted. However, in retail environments, which are generally a big extension of interconnected aisles, the time to complete an unassisted exploration is prohibitive. Compared to an unassisted exploration, the actual approach has two main advantages: optimizing the time it takes to recognize a zone; and empowering a user with an easy way to define the interesting zones for inventory. The latter, if doing unassisted exploration would require a posterior manual intervention or the addition of beacons for the robot to identify the interest zones. The ideal case would be that of a non human assisted exploration, but assisted by beacons or other technologies. Possible means for assisting explorations without human intervention include those discussed next for supporting localization.

At the moment, exploration is under testing, see Sect. 6.1. The exploration is being developed with the combined aim of granting AdvanRobot enhanced autonomy and producing 3D maps.

*Localization Robustness.* AdvanRobot requires a good accuracy in localization in both the *Recognition stage* and the *Inventory stage* in order to complete its tasks. During the *Recognition stage* AdvanRobot does not know the map of the environment, hence, it is executing a SLAM algorithm. It has been noticed that in very regular environments and large open spaces the localization of AdvanRobot is not reliable enough to generate faithful maps. Moreover, at the

<sup>19</sup> [http://wiki.ros.org/ar\\_track\\_alvar](http://wiki.ros.org/ar_track_alvar)



beginning of the *Inventory stage*, AdvanRobot needs to deal with the problem known as kidnapping [3]. To cope with this issues landmarks can act as absolute positioning references. Accordingly, the actual implementation makes use of QR codes. However, the detection of QR codes relies on a direct line of sight and good lighting conditions. Alternative means to support localization include laser reflectors, Bluetooth beacons and Battery-Assisted Passive (BAP) RFID tags.

*Robustness to layout changes.* Significant layout changes (or the accumulation of minor layout changes) can impact significantly the performance of AdvanRobot. In front of such changes AdvanRobot may not be able to reliably localize itself in the environment and fail to complete a mission. When this happens, not only the mission failure consequences have to be assumed but also the *Recognition stage* needs to be rerun. An interesting challenge is granting AdvanRobot with the capability of modifying the maps and goals of a zone at the same time it is running the *Inventory stage*. Hence, after several inventory iterations of a zone there would be no divergence from the original observations to the actual layout, which is the case at the moment. In practice, at every *Inventory stage*, the zone's *Recognition stage* observations would be updated and the impact of cumulative layout changes minimized. This would be equivalent to an assisted exploration, being the assistance the previous observations of the zone. In this way, there would only exist the need for a very first human assisted *Recognition stage*.

*Inventory and Location Navigation Strategies.* At the moment, the navigation is optimized for the compromise between time and inventory accuracy. However, a precise RFID location requires constraints to be met in terms of navigation [11]. Combining inventory and location constraints in a single navigation strategy is one of the main challenges to navigation control. After, it follows the addition of constraints for a complete 3D mapping of the environment. Combining optimally the constraints for inventory, location and 3D mapping would optimize the valuable outputs of AdvanRobot and, at the same time, minimize the time invested in the commission.

### 3.4 Human-Robot Interaction

Human-AdvanRobot interaction mainly consists of two operational procedures that simplify the user experience and minimize human errors. Both operational procedures are guided and executed by means of a control interface described next.

The first procedure empowers the user to launch a *Recognition stage* in order to create a map of the environment and get a set of indicative goals to follow when doing inventory. The second procedure lets a user launch the inventory of a sequence of selected zones, called *Inventory mission*. The second procedure can be scheduled and managed remotely.

The specific challenges and specifications related to human-robot interface are explained. AdvanRobot is a system that is specially suitable for large stores. Nevertheless, taking inventory of the whole store in a single mission is not always

possible due to time constraints. Moreover, the user might request to take the inventory of only a collection of specific products. To cope with this, two key aspects of AdvanRobot operations are introduced:

- The division of the shop floor.

The shop floor is separated in zones complying with the following:

- Zones should contain a family or a set of related families of products.
- Zones should be between  $750\text{ m}^2$  and  $1500\text{ m}^2$ .
- Zones should encompass easily identifiable architectonic features. For instance, it is no recommendable defining a zone as an island of hangers in the middle of a store. This is due to the problems that can arise in referencing the zone and the robot localization robustness.

Therefore, for each of the zones the robot keeps a separate map interlinked with the other zones' maps. The reason for this has been discussed previously in Sect. 3.3.

- Zones are identified using visual cues, in this case QR codes.

The linking and identification of defined zones is done using QR codes placed at the start and at the end of each zone. Noteworthy, the end of a zone is always the beginning of the following zone. Consequently, all the zones are interlinked and any sequence of zones can be selected for inventorying.

This two key aspects allow the user to easily select zones for inventorying and recognition. If the layout of the shop floor has changed considerably and the re-recognition of a zone is required, only the specific zone will need to be re-recognized saving AdvanRobot time as opposed to having to re-recognize the whole area (the sum of all the individual zones). And the user can identify such zone easily since it is marked at its beginning and end by a QR code.

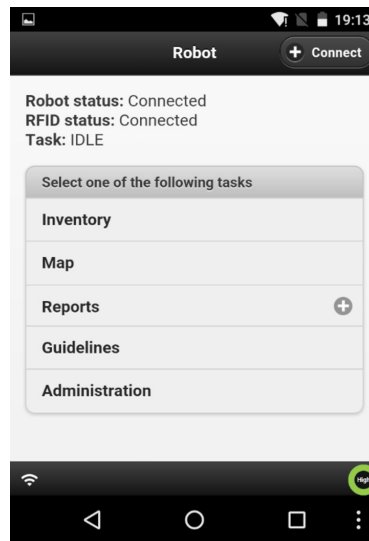
**3.4.1 Launching the *Recognition Stage*.** In order to launch the *Recognition stage* the user sequence of steps to follow is:

- Place the robot in front of the starting QR code of the zone. Doing so, AdvanRobot recognizes which zone is about to be recognized and informs the user on the interface for confirmation.
- By means of the human-robot interface the recognition is launched pressing the button *Start Recognition*.
- Guide the robot through the zone's interest spots, those intended for inventorying. While it is being guided, AdvanRobot records key spots where it is reading RFID tags. Later, the key spots are used to guide the inventory mission.
- When the RGBD camera detects the final QR of the zone, the interface pops up the options *Finish Recognition* and *Continue Recognition*. The first allows the user to end the process, the second is used to resume the guiding process in case, even the final QR has been detected, it is not yet over.
- If the option *Finish Recognition* is pressed, the map and key spots are stored and the *Recognition stage* processes stopped.

**3.4.2 Launching an *Inventory Mission*.** There are two procedures to start an *Inventory mission*. The first, which minimizes the user intervention, starts and ends AdvanRobot at its charging station. Given that the human-robot interface is a web application it can be accessed remotely. This empowers the user to launch *Inventory missions* programatically and remotely. The second is used when the user requires starting an inventory at a specific zone:

- Place the robot in front of the starting QR code of the first zone.
- Select the list of zones that AdvanRobot that comprise the *Inventory mission*.
- By means of the human-robot interface the Inventory mission is launched pressing the button *Start Inventory*.
- From this moment AdvanRobot is completely autonomous and its status and progress can be monitored on the human-robot interface.

**3.4.3 Human-Robot Interface.** The robot’s interface allows the user to interact with the robot in an intuitive and painless manner and it provides feedback of its status and progress. Figure 6 shows a snapshot of the human-robot interface.



**Fig. 6.** AdvanRobot interface

The interface guides the user along a sequence of steps that ensure the robot is prepared for the selected task. For instance, it gives guidelines to the user about the proper placement of the robot in front of a QR code, or it notifies the user to pull the e-stop button when it is needed.

Once all the steps have been successfully completed by the user following the interface guidelines, the specific selected task and its parameters are communicated to the *task\_manager* node (see Sect. 3.2.1) by means of ROS Services. During the task commission, the interface provides feedback by showing the progress of the task to the user. For instance, during the *Recognition stage* the interface shows the map together with the key spots that are being recorded; and when the robot is performing an *Inventory mission* it shows the progress of RFID readings and the progress of the mission itself.

In addition, the interface includes other relevant indicators. For instance, the snapshot of the interface, shown in Fig. 6, indicates that the robot is fully charged with the green circle on the bottom right corner and properly connected to internet with the white symbol on the bottom left corner. Also, it shows that the robot and the RFID systems are connected, and the robot status is IDLE, hence, any task can be triggered by the user.

The human-robot interface is a web application built using *HTML5* and *Javascript*. The communication with the ROS Master is accomplished using *rosbridge\_suite*<sup>20</sup>, a meta-package that provides the definition and implementation of a protocol for ROS interaction with non-ROS programs. *Rosbridge* is implemented using WebSocket as a transport layer and provides an API which uses JSON for data interchange. Finally, it also uses the ROS package *web\_video\_server*<sup>21</sup> for streaming the video of the RGBD camera.

**3.4.4 Challenges.** In a large retail store there are WIFI blind spots. Therefore, the connectivity with the robot through an infrastructure network can be lost unexpectedly. In case a user needs to interface the robot and the infrastructure connection is not available, without a backup connection the interaction becomes impossible.

In order to guarantee a responsive connection at any store location, the robot includes two wireless links: one as a client and the other as an access point. Usually, the robot is linked to the infrastructure network as a client, enabling its remote access and control. Moreover, the robot periodically uploads to the cloud relevant mission and status data, keeping a historical log that can be reviewed even when the robot is not online. In case the infrastructure network is not available in order to control or to know the robot status, AdvanRobot can be interfaced by means of its access point. As opposed to the infrastructure connection, this is available as long as the user stays within the robot's WIFI range. The roaming between links is performed automatically by the robot's interface, giving always priority to the robot's access point.

Hence, there is a valuable degree of redundancy in the robot's connectivity.

<sup>20</sup> [http://wiki.ros.org/rosbridge\\_suite](http://wiki.ros.org/rosbridge_suite)

<sup>21</sup> [http://wiki.ros.org/web\\_video\\_server](http://wiki.ros.org/web_video_server)

## 4 Simulation

The end-to-end simulation of the system has been set up using ROS and Gazebo. Simulation has been used for the validation of functionality in terms of navigation, operations and human-robot interaction. Yet, a realistic simulation of RFID is not available given its physical model is complex. RFID electromagnetic propagation suffers strongly from multipath effect, which means the RFID signal rebounds and is attenuated multiple times depending on the characteristics of the scenario. Modeling such behavior means taking into account each and every item and its characteristics within the reach of every single electromagnetic wave. At RFID frequencies not even raytracing produces satisfactory results. Only a full finite-element simulation of the entire environment, prohibitive from a computational cost point of view, would output reasonable simulation results.

Even though an RFID sensor is included for simulation in Gazebo, it is not implemented considering all the RFID simulation complexities. In conclusion, it is not possible with the available tools to simulate a realistic target scenario for the use case.

Accordingly, the simulation engine used for the RFID reads is not a physical but a probabilistic one. In this manner, only the throughput of RFID reads can be set to behave analogous to reality, which works for the validation of the coding of navigation strategies but not for the validation of their convenience regarding inventory accuracy and time. Hence, the validation and tuning of navigation strategies for an optimal compromise between inventory accuracy and time can only be performed in actual physical scenarios.

A set of packages for a basic simulation of the system can be found in the *UbiCALab* github repository<sup>22</sup> and a snapshot of the simulation is shown in Fig. 7.

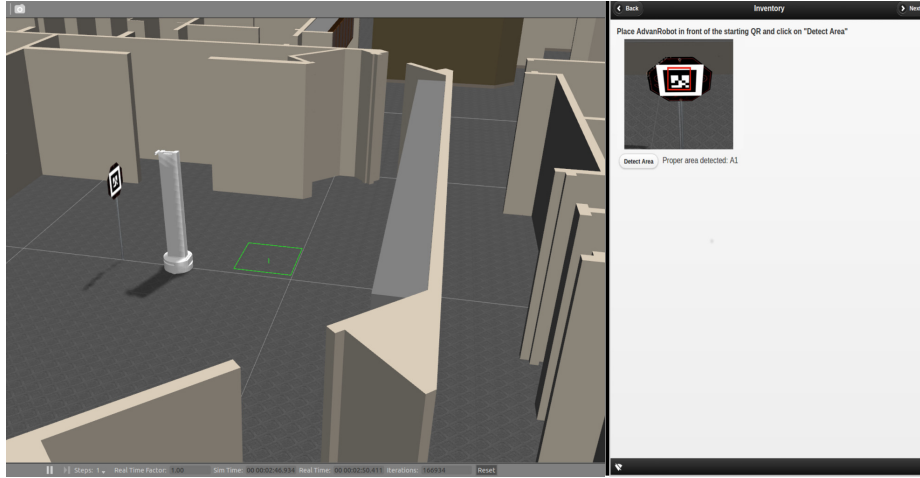
## 5 Experimental Results

AdvanRobot has been tested periodically in retail environments in every design iteration. The last version of AdvanRobot has been validated for a duration of 2 months at a retailer’s facility as the preparation for a subsequent pilot. The validation targeted AdvanRobot’s navigation on the shop floor; AdvanRobot’s RFID identification accuracy; and the operation by store associates after a training.

### 5.1 Navigation Validation

The navigation in a retail environment is not trivial due to the characteristics of shop floors. The main concerns at start have been the validity of the layout for a robust localization of the robot; floor materials and discontinuities; the ability to plan optimally paths; the effective speed of an inventory given the intricate configuration of aisles; and the effectiveness and negotiation of navigation commands. AdvanRobot has been in operation 8 hours per night for 40 nights in a

<sup>22</sup> <https://github.com/UbiCALab/advanrobot>



**Fig. 7.** Simulation of AdvanRobot at the moment of initiating an Inventory mission. On the left, a view of AdvanRobot standing in front of an initial QR. On the right, the corresponding view on the interface.

7500 m<sup>2</sup> store. Table 3 shows the maximum and minimum inventory times of the complete store throughout the iterations. The effective speed of AdvanRobot at inventorying is roughly 0.05 m/s, which is satisfactory but still the main figure to improve. The effective speed is compromised by the constraints of the RFID system and, at the same time, by the complexity of the layout for navigation.

**Table 3.** Maximum and minimum inventory times of the complete store throughout several iterations

Minimum time	23:41:33
Distance [m]	4,485
Effective speed [m/s]	0.052
Maximum time	31:50:51
Distance [m]	5,422
Effective speed [m/s]	0.047

None of the initial concerns were found critical in completing inventory missions. However, the robustness of localization is sometimes compromised by the lack of structural features to robot's observations reach. This is not a matter of installing more powerful sensors for location - a longer range laser - since it is usual for the robot on the shop floor to end up surrounded by expositor furniture. While the impact of this has not been critical to the day, it is considered a key aspect to improve. There are two main approaches to tackle localization

robustness. On one hand, improving localization algorithms. On the other hand, providing localization algorithms with additional observations of the environment.

## 5.2 RFID Identification Accuracy

Measuring inventory accuracy requires a baseline for comparison. The ideal case is that of manually counting each RFID label, which is known in retail as fiscal inventory. This seldom happens throughout a year given the workforce needed to count up to hundreds of thousands of items, a usual amount in a big retail mall. Consequently, a less demanding baseline in terms of man-hours is used. Currently, retailers use RFID handheld inventory devices for stock counting. Therefore, one of the references to compute the robot’s accuracy is the output of handheld devices. Moreover, retailers generally keep an inventory record which is an estimation based on items inputs and outputs but not on actual counts of stock on the shop floor. Even such kind of estimated inventory records diverge from reality quickly over time, they are still a good baseline for an arbitrated accuracy comparison. In conclusion, the two baselines that are used to measure the robot’s accuracy are handheld devices and estimated inventory records.

In order to measure the accuracy using the output of handheld devices, the baseline is computed as the sum set of items identified by AdvanRobot and by the handheld in a given zone. Table 4 shows results for a set of tests at selected zones. It is noticeable a higher AdvanRobot accuracy in all the compared cases. Interestingly, in the case of *Women’s wear* the handheld accuracy is significantly lower. Likely, the explanation for that is human error during handheld inventory taking. One of the main advantages of using a robot for inventory taking is preventing such oversights.

**Table 4.** AdvanRobot and handheld comparative accuracy

Product type	AdvanRobot’s accuracy [%]	Handheld accuracy [%]	Amount of RFID labels
Men’s wear	99.80	77.90	39,671
Women’s wear	99.90	44.10	22,277
Women’s underwear	99.54	72.77	8,778
Men’s underwear	99.43	88.91	1,055
Jeans	99.85	99.51	2,027

Besides, AdvanRobot’s accuracy was measured using an estimated inventory record of the whole retail store as baseline. In this case, the baseline itself is less accurate, which has an impact on the robot’s measured accuracy. One of the main reasons are items that are reported to be at the shop floor but are actually at back stores not visited by the robot. Note that estimated inventory records report stock keeping units (SKU’s) and quantities (count) instead of

unique item identifiers. This means that the comparison is not direct. Table 5 shows the accuracy of AdvanRobot measured at a store with more than 200.000 RFID labeled items. The column *Robot count matching* shows the count of product SKU’s identified by the robot matching the estimated inventory count. The column *Robot count in excess* shows the amount of references identified by the robot with a count higher than estimated. The accuracy measure is considered satisfactory by the retailer given the nature of the baseline, which is itself divergent from actual stock. Interestingly, the data set showed an excess of 19.938 references identified by the robot and not present in the estimated inventory record. This means that the robot identified product references that were not known to the estimated inventory record for some unidentified reason.

**Table 5.** AdvanRobot accuracy comparing to estimated inventory record

Estimated inventory count	Robot count matching	Robot count in excess	Accuracy [%]
209,465	162,335	12,025	78.7

In conclusion, the validation of RFID identification accuracy results successful in all the cases. Noticeably, AdvanRobot always outputs an accuracy above 99.5%. This is even more remarkable given the environment is highly dense in terms of RFID labeled products. In the exposed case there were over 200.000 RFID labeled items on a 7500 m<sup>2</sup> surface.

### 5.3 Operation by Store Associates

The suitability of the operational design and the usability of the human-robot interface is of utmost importance. For that, AdvanRobot was handed to store associates for its use for a month during the pilot’s preparation after a training. On the operational side it is noteworthy the users flexibility requirements. Each user has its own specifications depending on the details of the use case. For instance, the use of zones has proven useful in some cases while it was not necessary in others. A convenient approach is using a modular design ready for a quick customization. Since AdvanRobot is designed operationally based on a divide and conquer strategy, adjustments on the field are easy and quick to apply. A more complex question is human-robot interaction since potential operators include non-skilled associates. For that, the design of a user-friendly mobile app for AdvanRobot’s control and monitoring is crucial. We have noticed an initial steep learning curve mainly due to the lack of familiarity with advanced technology and a consequent low acceptance. Thus, while the interface has not presented remarkable issues, a good communication strategy to aid in the acceptance of a robotic solution on a shop floor is key for the success in its deployment.



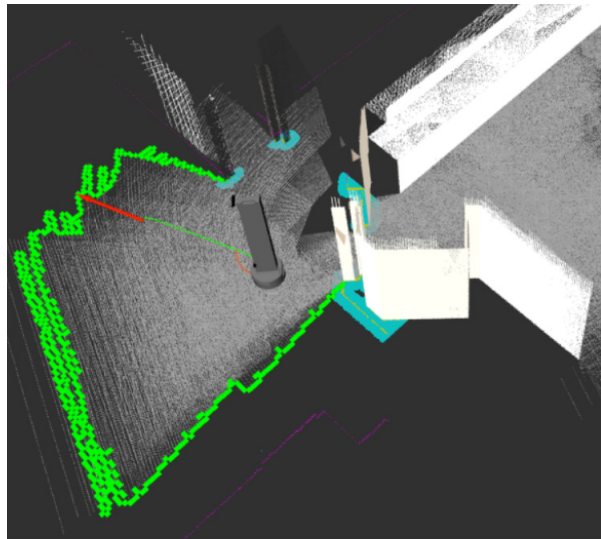
## 6 Ongoing Developments

### 6.1 Exploration for 3D Mapping

Building a 3D model of an environment can be done with robots equipped with RGBD cameras. Combined with the ability to locate products, it opens the door to a range of interesting possibilities such as measuring the impact of the placement of products, furniture and their combination in the sales; building a virtual store with the aim to link the offline to the online world; or verifying the layout, planogram and signage of a store. For this reasons, the generation of 3D maps is a use case of interest to potential users.

Currently, 3D mapping is working by means of the ROS package *rtabmap\_ros*<sup>23</sup> and the exploration for 3D mapping has been validated in simulation. Given the amount of factors that influence the output of an exploration, a ROS exploration framework has been developed in order to provide a fast and easy way of measuring the performance of different exploration approaches.

With the aim of generating a complete 3D map of the environment it is required sweeping all the space of interest. For that, a proper exploration strategy has to be applied. By using a 3 dimensional exploration, it is assured the completeness of observations of the space needed to generate the 3D map. Note that while the exploration considers the 3 dimensions of the space, the navigation is limited to 2 dimensions.



**Fig. 8.** Exploration testing using the Gazebo simulator. In green, the target frontier.

<sup>23</sup> [http://http://wiki.ros.org/rtabmap\\_ros](http://wiki.ros.org/rtabmap_ros)

A number of mature techniques exist tackling the robot exploration problem. One family is the frontier-based exploration which has long been exploited since its introduction in [18]. Frontiers are regions on the boundary between open space and unexplored space. By selecting a certain frontier as exploration target, the complete environment exploration is ensured. The exploration framework is intended for the frontier-based exploration technique.

The setup for the simulation using the developed framework includes two RGBD cameras. The second camera introduces an extra source of point clouds from a different perspective. Hence, objects are scanned from more than a single pose leveraging the 3D models. Furthermore, extra cameras are also beneficial for navigation purposes as the consequent increment of the field of view adds valuable redundancy to obstacle detection. The addition of more cameras is considered as they supply extra observation sources, beneficial for the completion and resolution of the 3D model.

**6.1.1 ROS Package: Exploration Framework.** As exploration for 3D mapping is a novel paradigm, the needs for versatility in testing different strategies is a key requirement. Hence, to choose the appropriate sequence of 2D navigation goals, a frontier based navigation framework has been developed as a ROS package (*cam\_exploration*), which is publicly available at UbiCALab github account, see fn. 4.

The main source of information used for the exploration is the projection of RGBD camera point clouds on the ground. This is achieved using *rtabmap\_ros*, a package based on the work presented in [10]. Basically, the package provides a whole SLAM implementation for point cloud data.

The data flow starts with the RGBD readings from the sensors which are published as ROS point cloud messages. These messages are used by *rtabmap* node to build a 3D model and to compute its projection on the ground as a map. This allows the differentiation of unexplored regions and explored ones. At this point, the *cam\_exploration* node uses the projection on the ground for exploration.

The *cam\_exploration* code structure is shown in Fig.9 with the developed libraries it contains. All the map related information is handled by the *map\_server* library. The node also provides visual information of its state using markers, which are handled by the *marker\_publisher* library. To keep track of the robot location and handle the interaction with the *move\_base* node, the *robot\_motion* library is used.

An important feature of this framework is its modularity. The main strategic decisions of an exploration that can be configured are:

- **Replanning.** To decide whether to send a new goal for exploration, a set of replanning conditions are used. Each condition represents a situation when it is desirable to send a new goal. Such conditions can be combined between them, so that their combination is what actually determines whether to send the new navigation goal. The combination method is an OR operation, so if any condition is met, the robot should send a new goal. This prevents the

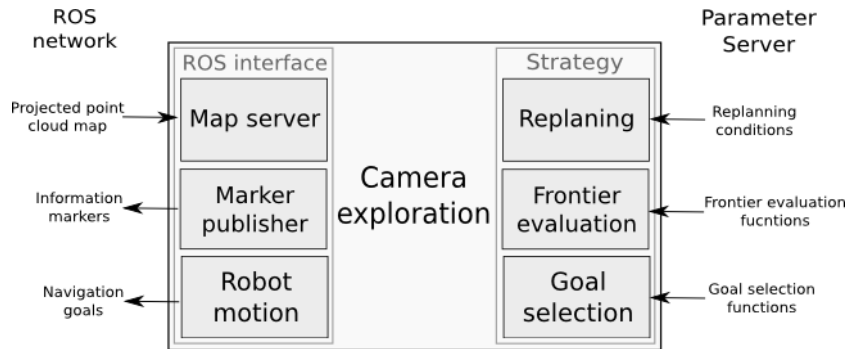


Fig. 9. Main *cam\_exploration* structure with its main libraries.

robot from getting stuck and takes profit of the information received while heading to the goal.

The current implementation includes the following replanning conditions:

- *not\_moving*: The robot is currently not heading to any goal.
- *too\_much\_time\_near\_goal*: Spending too much time near the goal. It votes for replanning if the robot has spent some time near its current goal and it is properly oriented with the goal.
- *isolated\_goal*: The goal is not close to any frontier. It is activated when none of the cells in an arbitrary large neighborhood of the goal corresponds to a frontier.

- **Frontier evaluation.** The main policy to be studied in frontier-based exploration is the choice of the goal frontier among the complete set of frontiers. To achieve that, a cost function is usually defined taking into account some criteria. Frontier evaluation methods are defined which can be combined in a weighted sum to provide the final evaluation. The implemented frontier evaluation functions are:

- Maximum size (*max\_size*). It favors larger frontiers over smaller ones.
- Minimum euclidean distance (*min\_euclidian\_distance*). It favors frontiers which are closer to the robot position, regardless of the obstacles in between.
- Minimum  $\mathcal{A}^*$  distance (*min\_astar\_distance*). The  $\mathcal{A}^*$  algorithm is intended to find a minimal cost path from a start point to a goal point in a grid. This optimal path is measured for each frontier from the robot's location and used as a distance measure. The function favors the frontiers with shorter distances in this sense.

- **Goal selection.** After a frontier is selected as the next exploration target, choosing a proper 2D navigation goal is not trivial. Specially, when working with projected point clouds. The actual 2D navigation goal is selected such that it is within the selected frontier and the robot cameras face the unexplored zone. The current implementation uses the frontier middle point (*mid\_point*). In practice, any function can be used to select the frontier point. A likely to consider function is the closest frontier point to the robot location.

All the options can be configured in the parameter server, so a simple YAML can be used to describe the exploration strategy in use.

**6.1.2 ROS API.** Following, the node main subscriptions, publications and parameters are described. Note that the actual 3D map is published by *rtabmap* node.

*Subscribed Topics.*

- **/proj\_map** (*nav\_msgs/OccupancyGrid*)  
Incoming map from *rtabmap* consisting in 3D camera point cloud projections.

*Published Topics.*

- **/goal\_padding** (*visualization\_msgs/Marker*)  
Region considered as the goal neighbourhood for robot-goal proximity purposes.
- **/goal\_frontier** (*visualization\_msgs/Marker*)  
Target frontier.
- **/goal\_marker** (*visualization\_msgs/Marker*)  
Selected goal point.

*Parameters.*

- **/cam\_exploration/frontier\_value/functions** (*list(string)*, default: [])  
List of frontier evaluation functions to be used. Possible values are *max\_size*, *min\_euclidian\_distance* and *min\_astar\_distance*
- **/cam\_exploration/<function>/weight** (*double*, default: 1.0)  
Value used to weight the function <function>.
- **/cam\_exploration/min\_euclidian\_distance/dispersion** (*double*, default: 1.0)  
Degree of locality of the function *min\_euclidian\_distance*.
- **/cam\_exploration/min\_astar\_distance/dispersion** (*double*, default: 1.0)  
Degree of locality of the function *min\_astar\_distance*.
- **/cam\_exploration/minimum\_frontier\_size** (*int*, default: 15)  
Minimum number of cells of a frontier to be considered a target candidate.
- **/cam\_exploration/goal\_selector/type** (*string*, default: "mid\_point")  
Way of choosing one of the target frontier points for target point. Only *mid\_point* is implemented
- **/cam\_exploration/distance\_to\_goal** (*double*, default: 1.0)  
Distance between the actual 2D navigation goal target frontier point. Should be close to the usual distance from the robot footprint to the nearest 3D camera point cloud projection point.
- **/cam\_exploration/replanning/conditions** (*list(string)*, default: [])  
List of replanning conditions to be applied. Possible options are *not\_moving*, *too\_much\_time\_near\_goal* and *isolated\_goal*.

- `/cam_exploration/too_much_time_near_goal/time_threshold` (*double*, default: 0.3)  
Maximum time in seconds allowed for the robot to be near a goal in *too\_much\_time\_near\_goal* replanning condition.
- `/cam_exploration/too_much_time_near_goal/distance_threshold` (*double*, default: 0.5)  
Minimum distance from the goal at which the robot is considered to be near it in *too\_much\_time\_near\_goal* replanning condition.
- `/cam_exploration/too_much_time_near_goal/orientation_threshold` (*double*, default: 0.5)  
Maximum orientation difference between the one of the robot and the one of the goal, to allow replanning in *too\_much\_time\_near\_goal* replanning condition.
- `/cam_exploration/isolated_goal/depth` (*int*, default: 5)  
Minimum rectangular distance from the goal to its nearest frontier allowed without replanning in *isolated\_goal* replanning condition.

## 6.2 Location of RFID items

Location of RFID labeled items is an actual topic of interest [12] and robotics contribution is paramount since it allows identification of items from multiple locations and knowing precisely the coordinates of that locations. Combining the latter with the detection model of the RFID sensor and applying proper probabilistic algorithms can output reasonable locations of the RFID labeled items. At the moment, an algorithm for the location of items is under validation. The estimated accuracy of the location algorithm is between 1 and 2 m, which is expected to be improved. The basic idea for the improvement of the accuracy is using extended detection instances and enhanced observations of the environment. For that, a precise location algorithm is being developed.

## 7 Future Work

In this section, a set of features in an exploratory or early development stage are discussed.

### 7.1 Collaborative Inventorying

The maximum area that the AdvanRobot can cover in a night shift is highly dependent on product density, and the complexity of the store layout. In sections with a lot of products per square meter AdvanRobot must slow down to allow enough time for the RFID system to read the thousands of tags that may be visible to the robot from a single pose. Another limiting factor may be sections with very narrow and/or irregular aisles, in which the effective speed of AdvanRobot is reduced.

As a result, each section of the store will require a minimum time for AdvanRobot to inventory. It may happen that a single robot is not able to inventory the entire store in a day. In this case there are two options: to complete the inventory in several days, or to employ a multi-robot network. Several robots may benefit from machine to machine communication to complete the inventory. This approach is not only more general and flexible, but also much more robust, as one robot may complete the job of another robot that might have malfunctioned or run out of battery.

## 7.2 UAVs and AdvanRobot Collaborative System

AdvanRobot achieves a 99.5% accuracy taking inventory in shop floors that are compliant with its navigation requirements. In order to extend the target scenarios, for instance to warehouses and distribution centers, and aiming at higher accuracy rates the collaboration with UAVs is considered.

When the AdvanRobot and UAVs will be working together, it is foreseen that the AdvanRobot will read most of the RFID tags due to its very high reading throughput, which makes it very efficient at inventorying dense environments. Oppositely, while a drone can reach places that AdvanRobot cannot reach, its reading throughput is much lower since the RFID system it can load and supply cannot be as powerful as that of the robot.

On the other hand, the UAV can inventory areas that are not accessible to the AdvanRobot: aisles narrower than 70 cm and shelves higher than 250 cm. Also, by observing the environment from a higher point of view, it can provide additional information to the navigation system for planning and exploration. Accordingly, the overall mission efficiency can be increased.

In addition, the robot can act as a charging station to UAVs. Usually UAVs have a limited autonomy due their limited payload capacity which implies low-capacity batteries. Hence, using the AdvanRobot as a mobile charging station can improve the operational availability of UAVs.

Table 6 summarizes the benefits of a collaborative system combining robots and UAVs.

**Table 6.** Impact of introducing UAVs in the system

	AdvanRobot UAVs Combined Impact		
Autonomy	High	Low	Mobile charging station.
Maneuverability [DOF]	2.5	6	World observations enhanced. Reading height > 2.75 m.
Passage width [cm]	> 70	$\leq 70$	Target scenarios extended.
Reading throughput	High	Low	Accuracy > 99.5%.

### 7.3 Applications derived from product location

The location of items on a map of the store enables the development of valuable applications both for customers and retailers. First, by knowing the location of an item it is possible to detect if this is misplaced. Item misplacement is a source of frustration for customers and implies a cost for retailers. An unknown misplaced item can be considered as stolen. Second, the location of items can be used to guide customers and associates to find easily a product or to produce an optimal path to find a set of products. This is commonly known as wayfinding and its output can reduce greatly the time needed for picking the products of orders placed online. Last, given the location of items, it is possible to analyze the profitability of products placements. For instance, a heat map of sales for a given product in different locations can be generated.

### 7.4 Simulation

Performing an end-to-end simulation of the system including the RFID propagation and detection model is a matter of interest. For the time-being, there is no Gazebo plugin for the faithful simulation of an RFID system due to its complexity. Thus, bringing forward RFID simulation in Gazebo is an interesting topic to work on in the future.

## Authors' Biographies

**Marc Morenza-Cinos** is a PhD candidate at Universitat Pompeu Fabra. His research interests include Robotics, Wireless Communication and Data Mining. Morenza-Cinos has an MSc in Information and Communication Technologies from Universitat Politcnica de Catalunya. Contact him at marc.morenza@upf.edu.

**Victor Casamayor-Pujol** is a PhD candidate at Universitat Pompeu Fabra. His research interests include Robotics, Artificial Intelligence and Aerospace. Casamayor-Pujol has a MSc in Intelligent Interactive Systems from Universitat Pompeu Fabra and a MS in Space Systems Engineering from Institut Suprieur de l'Aeronautique et l'Espace. Contact him at victor.casamayor@upf.edu

**Jordi Soler Busquets** is a Robotics MsC student at Universitat Politcnica de Catalunya. His academic interests include Machine Learning and Robotics. Contact him at jordi.solerb@upf.edu

**José Luis Sanz** is Industrial Designer at Keonn Technologies. His career has been developed between Conceptual Design and Industrial Design and development for manufacturing. José Luis has a degree in Industrial Design Engineering from Jaume I University in Castelln, a degree in Design from Universitat Politcnica de Catalunya in Barcelona and a posgraduate in Composite Materials from Eurecat Technological Center.

**Roberto Guzmán** owns the degrees of Computer Science Engineer (Physical Systems Branch) and MSc in CAD/CAM, and has been Lecturer and Researcher in the Robotics area of the Department of Systems Engineering and Automation of the Polytechnic University of Valencia and the Department of Process Control and Regulation of the FernUniversitt Hagen (Germany). During the years 2000 and 2001 he has been R&D Director in Althea Productos Industriales. He runs Robotnik since 2002. Contact him at rguzman@robotnik.es.

**Rafael Pous** is an associate professor at Universitat Pompeu Fabra. His research interests include Ubiquitous Computing, Retail Technologies and Antenna Design. Pous has a PhD degree in Electrical Engineering from University of California at Berkeley. Contact him at rafael.pous@upf.edu.



## Bibliography

- [1] Bertolini, M., Ferretti, G., Vignali, G., Volpi, A.: Reducing out of stock, shrinkage and overstock through RFID in the fresh food supply chain: Evidence from an Italian retail pilot. *International Journal of RF Technologies* 4(2), 107–125 (2013)
- [2] Durrant-Whyte, H., Bailey, T.: Simultaneous localization and mapping: part I. *IEEE Robotics Automation Magazine* 13(2), 99–110 (June 2006)
- [3] Engelson, S.P.: Passive map learning and visual place recognition. Ph.D. thesis, Yale University (2000)
- [4] EPCglobal: EPC Radio-Frequency Identity Protocols Generation-2 UHF RFID, Specification for RFID Air Interface, Protocol for Communications at 860 MHz 960 MHz, Version 2.0.1 Ratified (2015), [http://www.gs1.org/sites/default/files/docs/epc/Gen2\\_Protocol\\_Standard.pdf](http://www.gs1.org/sites/default/files/docs/epc/Gen2_Protocol_Standard.pdf)
- [5] European Commission: Proposal for a directive of the european parliament and of the council on the approximation of the laws, regulations and administrative provisions of the member states as regards the accessibility requirements for products and services (2015), <http://ec.europa.eu/social/BlobServlet?docId=14813&langId=en>
- [6] GS1: Regulatory status for using RFID in the EPC Gen 2 band (860 to 960 MHz) of the UHF spectrum (2014)
- [7] Hardgrave, B.C., Aloysius, J., Goyal, S.: Does RFID improve inventory accuracy? A preliminary analysis. *International Journal of RF Technologies: Research and Applications* 1(1), 44–56 (2009)
- [8] Heese, H.S.: Inventory record inaccuracy, double marginalization, and RFID adoption. *Production and Operations Management* 16(5), 542–553 (Sep 2007)
- [9] House of Representatives of the United States of America : Americans with Disabilities Act of 1990 (1990), [http://www.gs1.org/docs/epc/UHF\\_Regulations.pdf](http://www.gs1.org/docs/epc/UHF_Regulations.pdf)
- [10] Labbe, M., Michaud, F.: Online Global Loop Closure Detection for Large-Scale Multi-Session Graph-Based SLAM. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 2661–2666 (Sept 2014)
- [11] Miesen, R., Kirsch, F., Vossiek, M.: UHF RFID Localization based on synthetic apertures. *IEEE Transactions on Automation Science and Engineering* 10(3), 807–815 (July 2013)
- [12] NASA: RFID-enabled autonomous logistics management (realm) (rfid logistics awareness) (2016), [http://www.nasa.gov/mission\\_pages/station/research/experiments/2137.html](http://www.nasa.gov/mission_pages/station/research/experiments/2137.html)
- [13] Nur, K., Morenza-Cinos, M., Carreras, A., Pous, R.: Projection of RFID-Obtained product information on a retail stores indoor panoramas. *IEEE Intelligent Systems* 30(6), 30–37 (Nov 2015)

- [14] Rekik, Y., Sahin, E., Dallery, Y.: Inventory inaccuracy in retail stores due to theft: An analysis of the benefits of RFID. *International Journal of Production Economics* 118(1), 189 – 198 (2009), <http://www.sciencedirect.com/science/article/pii/S0925527308002648>, special Section on Problems and models of inventories selected papers of the fourteenth International symposium on inventories
- [15] RFID Journal: Tag-reading robot wins RFID journal’s coolest demo contest (2013), <http://www.rfidjournal.com/articles/view?10670>
- [16] Sarma, S., Brock, D., Engels, D.: Radio frequency identification and the electronic product code. *IEEE Micro* 21(6), 50–54 (Nov 2001)
- [17] Wang, J., Schluntz, E., Otis, B., Deyle, T.: A new vision for smart objects and the internet of things: Mobile robots and long-range UHF RFID sensor tags. *CoRR* abs/1507.02373 (2015)
- [18] Yamauchi, B.: A frontier-based approach for autonomous exploration. In: *CIRA*. pp. 146–151. IEEE Computer Society (1997)