UNIVERSITAT POMPEU FABRA
Department of Information and Communication Technologies

RFID autonomous robot for product inventory and location

Marc Morenza-Cinos

DOCTORAL THESIS
PhD Programme in Information and Communication Technologies

THESIS SUPERVISOR:
Prof. Rafael Pous Andrés

Barcelona, September 2018

To all those that do not get an opportunity

Acknowledgments

First of all I would like to thank professor Rafael Pous Andrés for the opportunity, guidance and enthusiasm throughout the development of this thesis. Also, I would like to thank all the Department members and the University staff for making every day an enriching and joyful experience. Special thanks to the colleagues with whom I have shared closely the day by day, either in teaching assignments or collaborating within the Ubiquitous Computing Applications Laboratory (UbiCALab) research group. Above all, my colleague Víctor Casamayor Pujol whose support and criticism was extremely valuable and, even more important, an essential companion during the long periods and endless nights spent testing the robot far from home.

I would like to thank Keonn Technologies for providing me with hardware and resources, trusting blindly in this project and my work. I would like to remark the involvement of José Luis Sanz, who has worked very hard and wisely in the industrial design of the robot and, from time to time, surprised us with tasty goodies from his land. Many others in Keonn Technologies deserve a big thanks: Moltes gràcies companys!

The Pompeu Fabra University Library has offered us the best possible testing environment and the kind collaboration of their professionals. My warmest thanks to Anna Magre, Ana Baiges, Natalia Plancheria and the rest of you.

A number of retailers have granted us with the opportunity to use their facilities for testing. It has been a piece of experience going here and there carrying the robot and meeting with many interesting people of different nationalities and backgrounds. Thanks to all of them for making my work less of a strain and more of an adventure.

Best thanks come in the end:

Gràcies a la meva mare i al meu pare per tot que no és poc. A la Núria, el Ferran, la Beth i la Noa pels ànims, forces i alegria que sempre em transmeten. Gràcies als meus amics per ser-hi sempre i fer-me d'aixopluc sota els xàfecs. El mode se atormenta una vecina acaba aquí.

I a la Glòria pel seu acompanyament incondicional. Això és molt teu.

Abstract

A solution for the automation of inventory taking and location of products in a store or warehouse is presented. Radio Frequency Identification (RFID), an automatic identification technology, and mobile robotics are combined in the design of an inventory robot. The navigation of the robot is commanded by an algorithm that takes as input the progress of new identifications. Such algorithm is essential for the robot to deliver an accuracy higher than 99% and for an optimal inventory duration. An interface for the interaction with the robot and a set of procedures for its operation are implemented. The location of items is implemented using two different approaches. The first approach applies clustering to streams of identifications and assigns the known location of a reference item to all the members of a cluster. The second approach applies Bayesian Recursive Estimation after the computation of an identification model. A methodology for the assessment is proposed and the data set generated for the analysis shared openly. Inventory accuracy and location are assessed in real scenarios. The proposed solution is demonstrated valuable and ready for the market.

Resum

Es presenta una solució per a l'automatització de l'inventari i la localització dels productes de tendes i magatzems. Radio Frequecy Identification (RFID), una tecnologia d'identificació automàtica, i la robòtica mòbil es combinen per dissenyar un robot per a inventaris. La navegació del robot està comandada per un algoritme que escolta el progrés de les noves identificacions. L'algoritme és essencial per tal que el robot obtingui una exactitud superior al 99% i per tal que la duració de l'inventari sigui òptima. S'implementen una interfície d'interacció i el conjunt de procediments necessaris per a operar amb el robot. La localització dels productes s'aborda de dues maneres. La primera consisteix en aplicar clústering a les cadenes d'identificacions dels productes i després assignar la localitzacio coneguda d'un producte de referència a tots els membres del clúster. El segon mètode de localització consisteix en aplicar Bayesian Recursive Estimation després d'haver computat un model d'identificació. Es proposa una metodolgia per a l'avaluació dels inventaris i el dataset generat per a l'anàlisi és compartit obertament. L'exactitud dels inventaris i la localització s'avaluen en escenaris reals. Es demostra que la solució proposada és de valor i està llesta per entrar al mercat.

Contents

Acknowledgments **i**

Abstract **iii**

1 INTRODUCTION **1**
1.1 Problem statement 1
1.2 Motivation 2
1.3 Contribution 3
1.4 Organisation 4

2 BACKGROUND **7**
2.1 Inventory record inaccuracy 7
2.2 Radio Frequency Identification 12
2.2.1 Inventory taking 14
2.2.2 Inventory location 15
2.3 Autonomous mobile robotics 17
2.3.1 Robotics and RFID 17

3 PRELIMINARY ROBOT DESIGNS **19**
3.1 The proof of concept 19
3.1.1 Preliminary results 21
3.2 The preliminary prototype 22
3.2.1 Learnings 23

4 ROBOT DESIGN **25**
4.1 Requirements 25
4.2 Robot hardware 28
4.2.1 Identification subsystem 28
4.2.2 Robotic subsystem 31
4.2.3 Interconnection 35
4.3 Robot logic subsystem 36
4.3.1 Navigation 36
4.3.2 RFID-driven navigation control 41
4.3.3 Mission manager 44
4.3.4 Task manager 46
4.4 Human-Robot interface 47
4.4.1 Recognition and inventory 48
4.4.2 Reports 52
4.4.3 Administration 52
4.5 Robot design validation 53

5 INVENTORY **57**
5.1 Methodology 57
5.1.1 Accuracy computation 57
5.1.2 Definitions 64
5.2 Verification and optimisation 70
5.2.1 Verification setting 71
5.2.2 Verification 74
5.2.3 Optimisation 78
5.3 Validation 81

6 LOCATION OF ITEMS **89**
6.1 RFID location dataset 90
6.2 Clustering identification streams 94
6.2.1 Similarity coefficients 99
6.2.2 Clustering 105
6.2.3 Clustering evaluation 107
6.2.4 Discussion 119

6.3 Recursive Bayesian estimation 122
6.3.1 Identification model 123
6.3.2 Experimentation 130
6.3.3 Conclusion 136

7 CONCLUSIONS AND FUTURE WORK 139
7.1 Conclusions 139
7.2 Future work 141

Appendices 143

A IDENTIFICATION STREAMS 145
A.1 Subset A 146
A.2 Subset B 148

B 3D MAP 151

List of figures 162

List of tables 164

References 172

Chapter 1

INTRODUCTION

1.1 Problem statement

The goal of this thesis is the development of a solution for the automation of inventory taking and the estimation of items location in stores and warehouses. An accurate inventory count and a good estimation of the items location open new opportunities to increase the operational efficiency of the supply chain and to enhance the customer experience. The solution proposed does not require either human intervention or the installation of any fixed infrastructure. In this sense, it is fully autonomous and space-agnostic. The solution uses Radio Frequency Identification (RFID), an automatic identification technology, combined with mobile autonomous robotics. While both technologies are mature and exploited separately in several applications, their combination for the automation of stock-taking had not been explored at the start of this thesis. In this sense, an algorithm that controls the navigation based on the progress of RFID identifications is required to yield an accurate inventory count. Regarding items location, algorithms are implemented and assessed in a real scenario. Figure 4.6 depicts the inventory and location robot designed and implemented in an actual retail store.

Figure 1.1: The robot taking inventory at an actual store.

1.2 Motivation

Commonly, inventory taking in a store or warehouse is performed manually by associates, which implies a great effort and unavoidable systematic errors. Humans performing repetitive, tedious and prolonged physical tasks are in nature fallible and, as a consequence, inaccurate. Moreover, such tasks cause injuries and dissatisfaction. As a consequence, brick and mortar retailers have no means to know what they have in their stores nor where they have it. This presents serious inconveniences for physical stores to converge with online commerce, which can offer immediate information regarding the availability of products. In fact, the strategical move of brick and mortar retailing is to offer the conveniences of e-commerce to their customers but inside the store. For instance, browse online and later try on or pick the order in the store. For that matter, the foremost challenge is ensuring that a customer’s order can actually be served. Failing to do so directly translates into customer frustration and a negative perception of the service. As a consequence, customers move to electronic and omnichannel platforms. Besides, an inaccurate visibil-

ity of goods within a supply chain hampers reaching optimal efficiency levels in operations, with a consequent loss of value. For instance, the reorder point of goods relies on estimated inventory levels. Therefore, an inaccurate inventory estimation may trigger unnecessary orders or fail to prevent stockouts. Furthermore, the inaccuracy is not only contributed by a deviation in an items' count but also by items' misplacement. An item that is not at the right location is in practice unavailable. Summing up, the lack of means to effortlessly keep an accurate and updated inventory record, including items' locations, hinders reaching a critical efficiency in supply chain operations to keep up with the pace of innovation.

1.3 Contribution

The topmost contribution of this thesis is a solution for inventory taking at retail stores. The solution is demonstrated to yield the most accurate to date stock count and location of items and enables updating the data without human intervention as often as once a day, as opposed to the current once a quarter or once a year.

At a lower level, an algorithm for the integration of the robot's navigation and the RFID data stream is proposed. A navigation control that takes as input the progress of RFID identifications is indispensable to maximize inventory accuracy. Additionally, a framework for a comparative assessment of inventory figures is defined. In reality, given that there are no accurate tools for inventory measurement more accurate than the proposed solution, it is very difficult to establish the ground truth. Hence, a procedure for accuracy evaluation is defined.

Regarding the robot's operation, a simplified set of procedures have been developed allowing its operation by unskilled users. Accordingly, a human machine interface for seamless control and monitoring of the robot has been designed. Both the operations and the interface have been verified and validated by real users.

For the location of items, two algorithms have been explored. On one hand, clustering has been applied to streams of identifications in order to

group items by spatial proximity. Reference objects with known locations determine the location of grouped items. In this case, item locations are discrete since they are those of the reference objects. On the second hand, Bayesian updating has been applied following a supervised measurement of the detection model. In this case, computed locations are continuous and reference objects are required only during the model measurement.

The dataset for the assessment of location algorithms is shared for reproducibility and future improvements. A test setting was prepared with the specific purpose of location assessment. It involved coding and placing RFID labels in 7000 books at the Pompeu Fabra University Library. The dataset includes thousands of observations recorded during several inventory rounds.

1.4 Organisation

This thesis is organised in seven chapters. Chapter 2 presents the context of this work, including the origin and consequences of Inventory Record Inaccuracy (IRI); the foundations of Radio Frequency Identification (RFID) and its application to inventory taking and location; and an overview of robotic technologies with a focus on solutions that combine robotics and RFID. Chapter 3 introduces early attempts to the design of the robot. It includes the initial conceptualisation of the idea and learnings from preliminary experimentations. The focus is on two designs, one that makes use of a humanoid robot, and another that leverages an old mobile base. These were the seeds of the final solution, presented in Chapter 4, where the design of the robot is explained, detailing its three main parts: the identification subsystem; the robotic subsystem; and the human-robot interface for operation. A study with real users is undertaken in order to validate the design. Chapter 5 focuses on the assessment of the accuracy of the proposed solution. A methodology is proposed, which includes the computation of accuracy and the definition of specific figures of merit. The actual experiments undertaken in real scenarios, University Library and a retail store, are described and their assessment completed. The

analysis includes the comparison of the robot's performance with manual inventorying. In Chapter 6 the algorithms to estimate the location of items are addressed, including the description of the location dataset prepared for the evaluation. Two location algorithms are described and their accuracy analysed. To conclude, Chapter 7 presents the conclusions and future work of this thesis.

Chapter 2

BACKGROUND

2.1 Inventory record inaccuracy

Inventory, in its broadest meaning, is defined as a complete list of items. In this sense, in a retail environment, this includes not only goods available for purchase by customers but also furniture and devices used for store operations. However, inventory is sometimes referred to only as the merchandise held in stock. Among several dictionaries and references, inventory is found to cover different meanings. What is clear is that disciplines related to controlling and managing the raw materials and goods in logistics use the term inventory: Inventory Control and Inventory Management. Hence, it is reasonable to deem inventory in its broadest meaning if not explicitly stated otherwise. Algonside inventory, the word stock is often found. Definitions and experts point that stock are items available for sale or distribution. As opposed to the broadest meaning of inventory, this does not include, for instance, assets. Nevertheless, in the literature inventory and stock are often used interchangeably. Given the nature of this work, inventory and stock are used equivalently under the definition

"The goods or merchandise kept on the premises of a shop or warehouse and available for sale or distribution." [1]

[1] https://en.oxforddictionaries.com/definition/stock

Inventory is one of the main aspects involved in the performance of supply chain logistics management. From an operational perspective, the supplier, enterprise and customer relation is built upon two links. On one hand, the inventory flow is concerned with the movement and storage of finished and unfinished goods. On the other hand, the flow of information provides visibility of the location and status of inventory and other resources. And visibility is the main driver of supply chain planning, similar to a sense-think-act network. For instance, guaranteeing an item availability requires knowing its actual updated position (visibility of inventory level) following its consumption, demand and replenishment. The integration of logistical operations involves, among others, the objective of inventory reduction. The specific objective is reducing and managing inventory levels to the lowest possible without compromising the overall supply chain performance. The reason is that keeping and managing unefficiently and un-effectively inventory translates into critical costs. For instance, an excess of moving inventory implies unnecessary transportation and, once arrived to the destination warehouse, unnecessary storage space. Several inventory policies exist with the aim of yielding a minimum service level without incurring in an unnecessary waste of resources. Most, if not all of them, rely on measured or estimated inventory levels to answer questions such as *when to order* or *how much to order* [1]. Like in every other system, uncertainties and flaws occur and inventory policies and their implementation, known as Inventory Management, are designed to counter them. For instance, while cycle stock are the amount of items needed to fulfil the expected demand, safety stock are the amount of items needed to fulfil deviations from the expected demand. Both are computed to offer a given service level and both rely on an alleged actual stock, since stock is updated incrementally [2].

An enterprise inventory exists in two different dimensions. On one hand, the actual quantity of goods or materials on hand, referred to as physical inventory. On the other hand, the amount of items or materials listed as available in the system used for monitoring inventory levels. The inventory monitored by a system is known as system inventory record, perpetual inventory or recorded inventory. Noteworthy, the recorded in-

ventory is an alleged list of items that usually, and in some cases notably, differs from the physical inventory. Raman et al., in an investigation of 370,000 SKUs at a leading retailer, found that 65% of the inventory records were inaccurate with an average discrepancy of 3.5 items [3]. Kang and Gershwin found similar figures. The average inventory accuracy measured across several stores was 51%. Moreover, one out of four SKUs differed from the physical stock by more than 5 items [4]. In both cases, the use of information technology to support replenishment was extensive. Kök and Shang report that 1.6% of items showed wrong inventory records even after carrying out several improvement programs to mitigate errors [5].

Inventory record inaccuracy (IRI) is defined as the discrepancy between the actual physical physical inventory and the recorded inventory [6]. IRI is considered one of the main causes of uncertainty and performance deterioration in the supply chain [4, 7, 8]. The consequences of IRI on a supply chain that uses information exchange for planning are interrelated. On one hand, unexpected stockouts may happen due to order placement based on wrong inventory figures. Consequently, the service level is affected, compromising the customer experience [9]. On the other hand, an overall increase of inventory costs. For instance, overstocking and order placement in excess due to short system inventory figures imply an excess of inventory to be stored and transported [10]. And the final consequence of a suboptimal supply chain is a loss of profit. Raman et al. estimated a profit loss of 10% of the total profit due to IRI in a leading retailer [3].

The main causes of IRI are four. First, shrinkage errors are those originated by lost and stolen items either by workers or customers. Shrinkage is a consequence of an intentional activity that does not involve updating the inventory record accordingly. Second, transaction errors are wrong inventory record updates, or its absence. For instance, registering incorrectly new supplies or registering at checkout a wrong stock keeping unit (SKU). In this sense, transaction errors are originated unintentionally by workers' activity. Third, misplacement includes the items that are not available because they cannot be found. And fourth, supply errors are

those caused by wrong supplier deliveries. For instance, receiveing a different amount, different type or even not serving an order and validating it in the system on arrival. Although quantitative analysis for each of this errors are not found in the literature, some exist. Raman et al. found that 16% of the items listed as available in the recorded inventory could not be found [3], with a derived 25% of losses.

At an operational level, the direct measure to correct IRI are periodical physical counts, known as inventory alignment. However, the implementation of physical counts is costly and, to date, they are not performed often enough to eliminate the discrepancies consistently. A complementary strategy is preventing the causes of IRI and its inclusion in the forecasting of resource planning. Nevertheless, although prevention can help, some of its factors can not be controlled given the involvement of intentional and unintentional actions. Perfect counterfeit measures and flawless operations are simply not possible [11, 12]. Regarding modelling, including IRI as a variable in the methodologies has been studied and it has been shown that daily IRI variability makes it difficult to reach a satisfactory performance [13]. In conclusion, IRI corrections are the most reliable way of mitigating its impact in supply chain performance, and its periodicity is critical [14, 15]. Figure 2.1 summarises the main causes and consequences of IRI.

IRI corrections are usually performed manually by associates or external audit companies. They are undertaken either with the support of barcode readers, RFID readers or without a supportive technology. In most cases the purpose of inventory-taking is balancing accounts. For this reason, it is only undertaken once or twice a year and called fiscal or accounting inventory-taking. Contrarily, periodical stock-taking for the correction of IRI is not a widespread method.

Sahin et al. examined the benefits of using barcode technology for IRI correction. They found out that, even it helps, errors still occur given it is a manual process. Consequently, the inventory information discrepancies are not consistenly corrected [7]. A research field exists called Behavioural Operation Management, which studies the implications of managerial decisions at approaching operational problems. How-

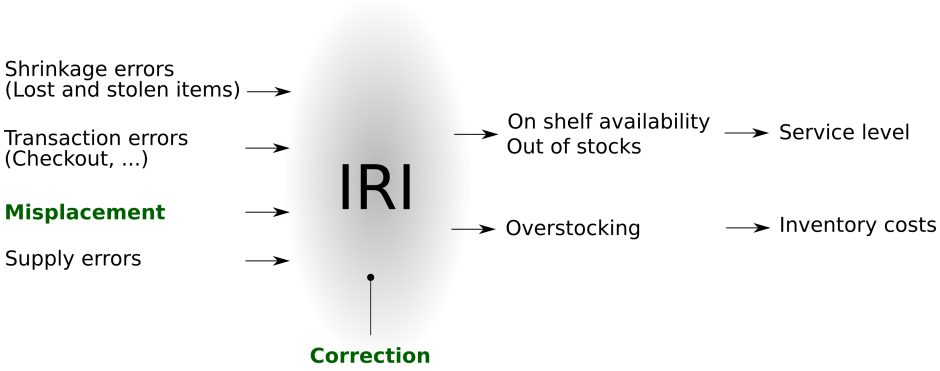


Figure 2.1: Summary of IRI’s main causes and consequences. Green coloured, the contributions of the proposed inventory robot to the mitigation of IRI. At the same time, it prevents one of the main sources of IRI and provides a means of a periodical and reliable correction.

ever, few works exist assessing behavioural aspects of the workers involved in the execution of supply chain tasks. Repetitive stock counting has been shown to decrease associates attention and, in turn increase their confidence on a resulting correct count [16]. More recently, cognitive fatigue as a consequence of repeated transactions has been empirically measured by using near-infrared spectroscopy [17]. From a psychological perspective, Einhorn and Hogarth studied how experience carries an associated excess of confidence that is evinced by fallible judgements [18]. In conclusion, human activity is error prone, more notably when tasks are repetitive and complex. And training does not suffice to avoid errors happenings. Moreover, it has been studied that the nature of retail staffing does not help in alleviating such errors. Due to retail seasonality part time employee hiring is usual. Chuang and Oliva analysed the contribution of part time employees in retail and concluded that they do not contribute to an enhancement of the job quality, IRI and service level [12].

In summary, while the correction of IRI is of utmost importance for the performance of the supply chain, manual alignments cannot grant target figures due to worker’s behavioural implications.

2.2 Radio Frequency Identification

Radio Frequency Identification (RFID) is an automatic identification technology that enables the simultaneous unique identification of hundreds of items without the need of a line-of-sight. In brief, a basic RFID system consists of an interrogator (or reader) and a set of transponders (or tags). The interrogator queries the environment by emitting a modulated signal through an antenna. The signal excites nearby RFID transponders that, in response, notify their unique identification number [19]. Transponders can be either powered by an embedded battery, active tags, or by the same signal emitted by the interrogator, passive tags. The latter imply shorter detection ranges but a much cheaper cost per tag. Thus, passive RFID tags are the ones to be used in applications that require labelling a massive amount of items. Barcodes have been the ubiquitous identification technology in the last decades. However, RFID offers indisputable comparative advantages. The most notable is that, as opposed to barcodes, RFID does not require a line-of-sight between the interrogator and the tag. Moreover, its reach can be of several meters, depending on the type and configuration. The RFID signal, similarly to the WiFi signal, propagates in several directions and can traverse different types of materials, being its reading reach up to tens of meters. The optical nature of barcodes does not make the same possible. Barcode requires a direct line-of-sight and a close, clear view of the label being read. A second advantge of RFID over barcodes is its capacity of identifying simultaneously hundreds of tags. Actual hardware [20] that complies with the actual protocol [21], if configured properly, can identify up to 750 tags per second. Therefore, RFID enables a bulk identification of items even if they are inside boxes or pallets. Interestingly, given the former advantages, RFID readers do not necessarily require of a person's involvement for operation. There is no more need for aligning a reader with a tag. Consequently, the reader can be fixed to the ceiling or a wall and any labelled item within identified automatically. No less important, RFID provides a unique identification of each item while barcode yields an identification at a product type level. This is the basis for bulk identifications, one does not need to worry about

having identified previously a similar object given there is no coincidence with other object's identifiers. Moreover, a fine grained traceability is enabled and the door to a thorough product life-cycle management opens. Let's take the case of fresh products. Using barcode the identification information would be *I am seeing a fruits yogurt of brand A*. Using RFID, the information would be *I am in front of a fruits yogurt of brand A that was produced day D1, in factory F and travelled in the refrigerated truck T to arrive in the store on day D2. Its expiry date is day D3 and ....* If the proper technology throughout the supply chain were deployed, virtually any question regarding the specific product could be answered. Last but not least, RFID tags are not only readable but also writable and can contain a significant amount of information compared to barcodes. RFID tags contain a memory chip that can be accessed and modified at any moment according to needs. For instance, an RFID tag can be used at the same time for item identification and for counterfeiting. A specific field in the memory is used for the product identifier along with a flag that is deactivated once it has been purchased. Additional information could be written in the RFID tag if needed, like the temperatures it has sensed in key spots during its life-cycle. Furthermore, being writable, RFID tags can be reused. In clothing retail it is common using hard tags, tags designed for durability, that are removed from clothes at checkout, rewritten and placed again in other clothes exposed to the public.

For all this reasons, RFID has been regulated and adopted as the item identification technology by an increasing number of leading retailers that seek to enhance inventory visibility and be more competitive in an omnichannel environment. Figures reported by industry stakeholders point to RFID as the adopted de-facto standard for inventory visibility. The state of RFID adoption in 2016 by top 100 US apparel retailers was surveyed by Auburn University RFID Lab [22]. It was found that, 35% of the retailers were conducting a proof of concept (POC); 22% were conducting pilots; 39% were in a partial deployment; and 4% already completed full deployment. More interestingly, there was a 32% increase of new adopters from 2015 to 2016, following a 23% increase from 2014 to 2015. A more general survey of 801 US small and large product manufacturers and retailers

found that in 2014 only 22% were familiar with RFID. Most of them in the apparel, footwear and related accessories industries. They reported that, on average, 40% of items are manufactured with tags and 47% of items received already have a tag [23]. Impinj, a leading tag manufacturer, reports having shipped 7.1 billion tags in 2017 [24].

2.2.1 Inventory taking

Following the adoption of RFID as an automatic identification technology a degree of technification in stock-taking has been introduced. Associates move with an RFID handheld interrogator and place it close enough (one meter) to batches of items for their identification. A visual, auditory or haptic cue is triggered when the RFID interrogator does not detect any new items. In this way, the associate knows when to proceed to a next batch of items. While this is a relevant step in stock-taking, it is still a manual procedure, which involves negative consequences. Stock-taking using RFID handheld interrogators is a repetitive and complex to plan task susceptible to human error, specially in large environments. For instance, it is usual that associates forget to scan an area or think they have already scanned it. The fact of being a manual repetitive and complex task is the cause of accuracies lower than expected.

Solutions for the automation of stock-taking on retail shop floors have been developed in recent years. To the best of our knowledge, during the development of this thesis, three commercial inventory robot made appearance. Two of them propose a solution similar to ours. StockBot [25] and Tory [26] are designed to automate inventory by exploiting RFID. The third solution is called Tally [27], which uses vision for shelf auditing. However, by using vision it is not apparent the identification at an item level. Unfortunately, there are no references in the literature regarding their design, operation and performance.

Few former works presented experimental solutions combining RFID with robotics for stock-taking. Ehrenberg et al. [28] presented a mobile platform equipped with an RFID reader that takes inventory and finds misplaced books at a library. The RFID technology used (HF) differs from

the de facto standard (UHF), the one embraced by the industry. In [29], a prototype of a robot that uses RFID to identify products in a mock-up of a supermarket is presented. RFID data is fused with vision and placed in a 3D model of the environment. Although the system is promising in the demo scenario, further experimentation and a detailed analysis of inventory figures are missing. In [30] RFID data captured with an early prototype of an inventory robot is exploited to create indoor enriched views of a store. Yet, performance regarding the inventory is not in scope. Zhang et al. [31] share experiments with a robotic inventory system on a mock sales floor. Results are encouraging but the amount of items is not comparable to a real store. In a prior work [32], we presented in detail the architecture of our autonomous inventory robot and an overview of accuracy results in a retail store.

Looking at solutions that propose a degree of technification at stock-taking using RFID devices, Bertolini et al. [33] compare the performance of RFID inventory counting to barcode counting using handheld devices at a real store. They conclude that RFID inventory is generally more reliable with accuracy figures ranging from 90.6% to 98.7% on a single scan. In [34], the performance of overhead RFID antennas installed on the ceiling of an actual retail store is compared to RFID handheld scans. The average accuracy of the overhead antennas is 93.0%.

2.2.2 Inventory location

Location of RFID labels has been widely studied from different perspectives in the last decade. Miesen et al. [35] classify the techniques upon the type of hardware and the RFID features used for location. In the backscatter domain and using commercial of-the-shelf devices (COTS), the case treated in this work, there are three possible approaches: Received Signal Strength (RSSI); Phase evaluation; and the combination of the former.

In [36], a detection model is bootstrapped and a particle filter applied for the location of items. A similar approach is used to map UHF RFID tags, computing a 3D sensor model that combines tag detection and RSSI [37] [38]. The best accuracy reported using this techniques is 0.202 m,

which is achieved applying several extensions to the original concept, for instance filtering of ghost-detections. However, there is no insight of its potential applicability to non-line-of-sight (NLOS) cases, where RSSI is known be radically affected and the RSSI-based model used could be not valid.

A Synthetic Aperture Radar (SAR) technique is proposed in [39]. A set of measurements are taken by a moving antenna and a spatial probability density computed, the holographic image, to assess the most likely tag location. The holographic image is generated by updating each pixel with the probability of a measure's received phase being originated at the pixel. The technique is similar to Angle-Of-Arrival (AOA) phase-based methods but with a probabilistic flavour. The main downward of this technique is that it requires taking a number of consecutive measures at a minimum of a fourth of the RFID wavelength. Otherwise, there is aliasing and the location cannot be accurately computed. Such constraint is not assumable by a solution designed for inventorying thousands of objects, given it would slow down unacceptably the task. Moreover, this technique is tag-orientation dependant and the tag orientation cannot be controlled in real scenarios.

Ma et al. [40] propose an algorithm that extends hyperbolic positioning based on phase measurements. It is similar to SAR in that the diversity of measurements is achieved by a moving antenna. The solution of the intersection of hyperbolas, the alleged tag location, is found by particle swarm optimization (PSO). The algorithm achieves a location accuracy under 0.2 m and is robust to tag orientation and noisy measurements. Authors point out that the technique may not be applicable to complex environments due to phase measurement errors.

BackPos [41] is a technique that introduces location of tags by solving the hyperbolic equation that relates the phase of the wave to the distance travelled. A system of such equations is solved applying constrains that take into account the directionality of detections. The mean accuracy reported, using simultaneously 4 antennas, is 0.128 m. Unfortunately, such technique relies on a LOS detection.

Although in practice, approaches that use non-COTS devices or spe-

cific system settings are not adaptable to the robot scenario, they are of interest and of potential inspiration.

Angle of activation (AoAct) technique [42] steers the interrogator beam to sense detection and non-detection events and determine de direction of the tag. It performs a triangulation of directions to determine the location of the tag. A 0.7 m accuracy is reported in a multipath environment. It requires a specific purpose antenna array and reader control.

In [43], the estimation of a tag's position is computed by applying a maximum likelihood estimation to a previously calibrated MIMO system that encompasses several antennas. Position estimates are under 0.1 m but, as aforementioned, the algorithm requires an initial calibration and a distributed antenna configuration.

2.3 Autonomous mobile robotics

2.3.1 Robotics and RFID

RFID has been widely used as a support technology in robotics. In [44] RFID is used on a robot to improve its global localization and reduce computational demands. RFID tags are used as features of a space in addition to its layout, which is captured by laser readings. [45] propose a localization algorithm that relies only on memorized snapshots of the detections of RFID tags deployed in the environment. A combination of RFID and vision is used in [46]. The gross location of a robot is determined using RFID, which identifies a whole area. After, vision is used to compute the fine pose of the robot within that area. Beside self-localization, RFID has been used for the identification and posterior manipulation of objects by a humanoid robot [47]. A similar use is described in [48], a robot identifies and finds objects that include RFID labels. And, in collaborative exploration, RFID tags can be used to mark already explored zones with information for other agents to gain the knowledge [49].

Chapter 3

PRELIMINARY ROBOT DESIGNS

3.1 The proof of concept

The first experiment combining RFID and robotics used an RFID system mounted on a structure and a wheeled humanoid robot. The structure included a pair of handles in order for the robot to hold it with its articulated hands. The RFID system was designed as an accessory for the existing robot, thinking of a scenario in which the robot would be multi-tasking and in any given moment could perform an inventory. The RFID system, shown in Figure 3.1, included 6 antennas and 1 RFID reader connected through 1 multiplexer.

Embedding 6 antennas allowed for the identification of items at heights ranging from ground level up to human-reach height. Given the robot was designed to work in spaces frequented by people, objects were not expected to be placed higher than at arms reach. The connection with the robot included a Power Over Ethernet (POE) link to feed the RFID system and, at the same time, control and gather the generated RFID data. At this stage, control involved starting the RFID system, asking periodically for inventories and stopping the system. More interestingly, the RFID system triggered a signal when it was not registering identifications from new ob-

(a) **(b)**

Figure 3.1: (a) Proof of concept of the mobile RFID system for inventory. Labelled objects were inside the boxes on the shelves. (b) Detail of the RFID subsystem that the humanoid robot is carrying.

jects. On the robot’s side, it was programmed to move forward one step at a time on a predetermined path. The step could be set to variable distances. After every step the robot would not move again until the signal informing of no new identifications was received. Such rudimentary form of RFID-driven navigation for inventory is detailed by Algorithm 1.

Algorithm 1 Rudimentary RFID-driven navigation control

```
1: D ← configured distance;
2: while inventory path not completed do
3:     identify items;
4:     if new items identified then
5:         in place angular movement;
6:     else
7:         move D centimetres forward;
8: inventory completed;
```

3.1.1 Preliminary results

This early proof of concept was intended to analyse the feasibility of automating inventorying. A set of labelled objects were placed on shelves, most of them inside boxes, and the robot moved in front of them to take the inventory. The items used for the analysis belong to six categories: apparel; books; perfumes in cardboard boxes (perfumes 1); perfumes wrapped in metallic foil (perfumes 2); DVDs; and CDs. Three tests were performed varying the navigation parameters and the results analysed. Table 3.1 summarises the tests and results.

Table 3.1: Results of inventorying with the preliminary humanoid robot design.

Product type	Number of items	Accuracy D=25cm	Accuracy D=15cm	Accuracy D=5cm
Apparel	40	97.5	100.0	100.0
Books	250	66.8	75.2	75.6
Perfumes 1	60	96.7	96.7	96.7
Perfumes 2	30	66.7	70.0	70.0
DVDs	50	-	100.0	100.0
CDs	200	-	100.0	100.0

Unsurprisingly, some types of items, namely those in higher densities or conformed partly by metal or liquid, showed lower inventory accuracies. Particularly books, which where packed inside a box and made of paper, which contains a considerable amount of water, and the perfumes wrapped in metal foil (perfumes 2). There was an expected increase of accuracy when reducing D from 25 cm to 15 cm, but negligible from 15 cm to 5 cm. This early results already point to the need of developing an adaptive navigation control.

3.2 The preliminary prototype

A preliminary prototype was constructed reusing a mobile base provided by a robotics partner[1]. The aim was to build a specific purpose mobile system to take inventory autonomously in stores. As opposed to the proof of concept, this would not need articulated limbs and complex humanoid robotics and it should be as simple as possible to undertake inventory taking. Not only its performance was to be analysed but also its eventual market viability. At this stage, the simplest mobile base included motors for navigation, a laser range finder for self-localization, an onboard PC for algorithm computation and a battery with enough power to feed both the RFID system and the base. Figure 3.2 shows the naked mobile base.

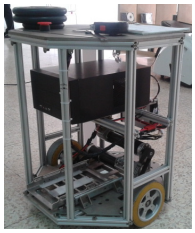


Figure 3.2: Naked mobile base used in the robot prototype.

With the base as a starting point, a structure to hold the RFID system was designed. This time, thinking of the spatial efficiency of identifications, two blocks of 6 antennas were placed, one on each side of the robot. In this manner, the robot could identify simultaneously items on both sides of an aisle while moving forward. Antennas were placed from ground level up to 2 meters to identify items up to arms reach. The resulting robot is shown in Figure 3.3. Note that the bottom aperture, the laser's window, constrained the laser's field of view to under 180º. However, its actual sensing angle range goes up to 270º.

The prototype had autonomous navigation capabilities. As opposed to the proof of concept, the robot would not follow a pre-defined path but could plan and self-locate on a pre-recorded map to reach a given

[1] https://www.pal-robotics.com

Figure 3.3: Robot prototype in a store.

goal. The RFID-driven navigation control paradigm was updated to work by interruptions to the navigation. At detecting a configured amount of new items, the underlying navigation control was interrupted and resumed when no new tags were detected. In this way, the fixed length step proposed in the proof-of-concept was substituted by an adaptive step determined by new tags detection.

3.2.1 Learnings

The preliminary prototype was brought to real scenarios in order to validate the feasibility of a robotic solution using RFID for inventorying. The preliminary measured accuracies, between 95% and 99%, were encouraging although limitations and room for improvement arose. First and most important, the robot’s environment observation capabilities. The prototype mounted a laser range finder, one that can sense a 2D slice of the environment. In this manner, it can sense the obstacles that appear at a fixed height (the laser’s height) but not obstacles at any other height. Consequently, the risk of crashing into obstacles that do not appear at laser’s height was a challenge. Additionally, the laser field of view is limited by the robot’s structure, which hampers its ability to perform

self-localization using a map since it can only use a partial view of laser scans.

A second important limitation was the lack of dampers, which hindered the robot’s ability to navigate on irregular floors. Actually, the prototype could not overcome small discontinuities such as a change of floor type from tile to carpet. In some cases, it even got stuck due to imperceptible height differences between tiles, compromising the wheels grip by the lack of a compensating force towards the floor.

An additional limitation was the robot’s speed, which slowed down the inventory and limited its manoeuvrability, creating a perception of clumsiness. Furthermore, an intuition from observing the prototype taking inventory was that the interruptions of the RFID-driven navigation control were too discontinuous. The interruptions were triggered based only on the last inventory observations, which could induce a high frequency of transitions from stopping due to new identifications to resuming navigation. Moreover, the robot was set to resume the navigation only when no new tags were received, which makes the robot too conservative thus slow at inventorying.

Lastly, there was room to improve the accuracy of the inventory, mainly by improving the RFID-driven navigation control strategy.

All this learnings were taken into account in the minimum viable design of the inventory robot and enhancements were introduced accordingly.

Chapter 4

ROBOT DESIGN

In this chapter a minimum viable design of the inventory robot is presented, from conception to implementation. The solution is built upon a set of requirements learned after experiencing with the preliminary prototype in real scenarios and by collecting observations from potential end users. The design proposed includes not only the hardware and application logics but also a human-robot interface that encapsulates the essential procedures for operation and monitoring. The ambition behind the inventory robot design is proposing a solution for inventory taking that can be deployed effortlessly in any retail store.

4.1 Requirements

The main purpose of the inventory robot is the automation of stock-taking at shop floors. Regarding automation, the main requirement is being autonomous and robust to changes in the working environment. On a store floor, while architectural features do not change and can be used as a consistent reference for navigation, periodical modifications of the furniture happen, which modify the layout. In addition, unexpected obstacles such as misplaced gear or persons are likely. Therefore, the navigation needs a degree of adaptability to deal with a changing environment. In brief, the robot cannot just follow a predefined path, as it would go on rails,

since it is intended to work in changing environments. Regarding stock-taking, the critical requirements are accuracy and time, which represent a design tradeoff. The question is how to use RFID on an autonomous mobile agent and guarantee accuracy in the shortest possible time. In this regard, the interaction of the two subsystems, the RFID system and the autonomous mobile agent, needs to be addressed. In brief, the robot should stay longer where products are numerous, and can move ahead if there are no items or those have already been counted. A third broad requirement is that the robot must be usable by persons without specific technical skills. Even though the robot is autonomous after an initial configuration, some procedures need to be assisted by persons. Moreover, without an interface for interaction, the robot cannot be monitored, which is critical for acceptance and integration into store operations. Accordingly, procedures for robot operation are defined, and a user interface for control and monitoring implemented. In this way, the robot should be manageable by any person able to use a smart phone.

In summary, the inventory robot is designed to meet the following requirements:

a Minimal inventory accuracy higher than 99%

b Inventory duration equal or less than current stock-taking procedures (usually handheld RFID readers)

c Autonomous navigation in changing environments

d Ease of operation and servicing

e Ease of assembly and transportation

Location of items can be estimated if a diversity of identifications in space is available. Given that the robot is designed to navigate the space while taking inventory, such assumption is very likely. Accordingly, location is not explicitly included as a design driver in this chapter. Nonetheless, the location of items is addressed thoroughly in Chapter 6 and the former premise proved.

The robot is built composed of three subsystems, which are closely related to the requirements. First, the identification subsystem, which is built using commercial off the shelf (COTS) RFID hardware manufactured by Keonn Technologies[1]. Second, the robotic subsystem is extended from a commercial hardware and tuned according to the requirements to provide a safe and robust autonomous navigation. And third, a human-robot interface enables the minimum set of operations conceived for control and monitor. Although subsystems and requirements are related, their relation is not univocal. On the contrary, the requirements are met by exploiting subsystems collaboration. For instance, if the robotic subsystem does not take into account the progress of identifications, the inventory accuracy can be seriously compromised. Note that although RFID has the capability of identifying simultaneously a number of objects, identifications in the presence of a considerable amount of items require more time. For this reason, the navigation should take into account and use as input the progress of new identifications. Table 4.1 summarizes the contribution of each subsystem to the requirements, which is further explained in this section.

Table 4.1: Contribution of subsystems to requirements

	Identification subsystem	Robotic subsystem	Human-Robot interface
Inventory accuracy	x	x	
Inventory duration	x	x	
Autonomous navigation		x	x
Operation and servicing	x	x	x
Assembly and transportation	x	x	

[1] https://www.keonn.com/

4.2 Robot hardware

This section gives an overview of the robot hardware, which consists of two independent parts: the identification subsystem and the robotic subsystem. The reason to treat them as two independent interconnected parts, as opposed to a one-body device, is modularity. Modularity is an essential characteristic aiming at an ease of servicing and transport, two of the design requirements. A "vehicle plus payload" model was used in the design.

4.2.1 Identification subsystem

The identification subsystem uses radio frequency identification (RFID) for the automatic identification of objects. The main components of a typical RFID system are RFID readers and RFID antennas. The readers include active RF electronics that excite antennas for the propagation of the RFID signal. Generally, a reader is used in combination with multiple antennas distributed strategically in space to maximize the volume covered by the RF signal.

The reader used is Keonn's AdvanReader-150[2]. It is a self-contained reader which encloses a single-board computer for control and external systems interfacing. The reader allows the simultaneous connection of 4 antennas. The antennas chosen, given their form factor, high gain and appropriate radiation pattern are Keonn's Advantenna-p22[3]. The radiation pattern determines the volume covered by an antenna's signal.

The first premise regarding the design of the identification subsystem is that it should be able to identify all the objects in scope. While the robot's mobility grants a complete coverage on the xy-plane, as long as the navigation is well-planned, the identification subsystem construction needs to address coverage also in height. Since the robot is intended for stores where products are placed at most up to customer's reach, the identification subsystem is designed to identify RFID tagged objects from the

[2]https://www.keonn.com/rfid-components/readers/advanreader-150.html
[3]https://www.keonn.com/rfid-components/antennas/advantenna-p22.html

ground level to 2.5 meters, which is achieved by placing several antennas along the z axis.

The second premise is a high accuracy, which requires spatial redundancy in the identification subsystem. This means that the antenna radiation patterns should overlap, both simultaneously and as the robot moves around the space. This is justified in prevention of two common effects in RFID systems that compromise accuracy: blind spots and occlusions. On one hand, blind spots happen where multipath reflections result in a cancelled or very weak wave. On the other hand, occlusions happen when the signal hits an RF reflector, which is equivalent to blocking the signal. It is common that occlusions happen due to the presence of metallic objects as well as due to RFID tags next to each other. Both effects are a threat to accuracy, more notably in environments crowded with RFID tags. In order to minimize these effects, the subsystem is designed with overlapping radiation patterns. In practice, antennas are placed close enough for an overlap of their radiation patterns. The resulting antenna configuration consists of 6 antennas placed next to each other along the z axis up to a height of 2.5 meters. Redundancy also occurs as the robot moves and its antennas cover a volume previously covered from another position. In this sense, the mobility of the robot also contributes to yield the required accuracy.

The third premise is that in order to minimize inventory time, identifications need to be parallelized. Parallelism is achieved by installing several RFID readers, each one in charge of a subset of antennas. In this way, since the RFID protocol allows for simultaneous signal transmissions, there can be simultaneous identifications. At the same time, the subsystem includes two sets of antennas, each one covering the two opposite directions, perpendicular to the robot's forward movement. In this way, objects on both sides of an aisle can be identified at just one aisle traversal. Otherwise, installing a subsystem that would cover a single direction, the aisle should be traversed twice and the inventory duration would be double. Note that on a shop floor, the usual layout consists of corridors delimited by shelves on either side loaded with products.

To sum up, the identification subsystem consists of 12 RFID antennas

distributed in two blocks of six antennas on opposite directions. Three RFID readers control four antennas each. In this way, the usual 400 identifications per second that one reader can process is multiplied by three. Figure 4.1 depicts a schematics of the identification subsystem and its parts.

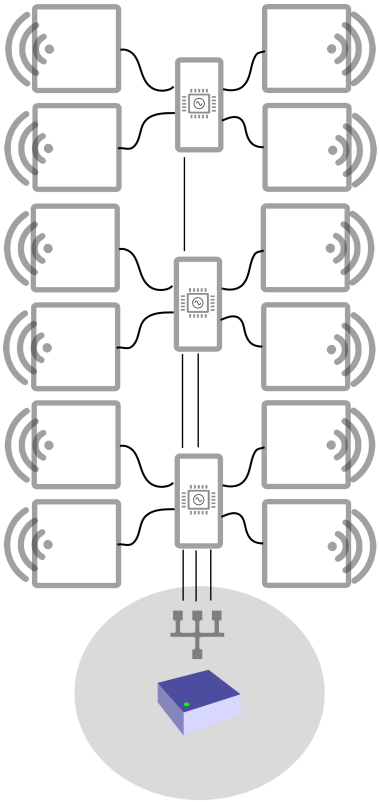


Figure 4.1: The identification subsystem consists of 12 antennas fed by 3 independent RFID readers. The identifications are gathered by a central process through an Ethernet link.

In what regards ease of operation, servicing, assembly and transportation, the identification subsystem structure takes them in consideration. An important design decision is making the structure foldable, which makes the robot more manageable for operation and transportation given

its height. Additionally, electronics are embedded in the structure in an accessible and modular way for the ease of servicing.

4.2.2 Robotic subsystem

The robotic subsystem is a customization of the commercial mobile base RB-1 by Robotnik[4]. The mobile base is differential driven, it includes two independent motor wheels that allow for a zero radians turning radius. This implies it can rotate in place, thus face any direction without changing position, which is of utmost importance to manoeuvre in narrow and intricate store layouts. The mobile base diameter is 50 cm, allowing for navigation in aisles as narrow as 70 cm, the measure established internationally for wheelchair accessibility. Each motor wheel is attached to a damper, which is essential in order to overcome floor irregularities and discontinuities as well to deal with the momentum at accelerations and brakes. Three omni-directional wheels act as casters for stability. Figure 4.2 shows schematics of the mobile base.

The robotic base includes a set of sensors for navigation and additional essential hardware.

Sensors

First, a laser range finder, is necessary to feed the algorithms that compute the map of the environment and the location of the robot on such map during navigation. Although a three-dimensional laser would be very appropriate, prices are prohibitive. Hence, a two-dimensional laser range finder is chosen. The laser range finder model is chosen considering the requirements of the algorithms for mapping and location. These are: a high precision of measurements; a high update frequency; and a considerable range and field of view. Additional considerations are suitable data and power interfaces, an Ethernet link and an open-end cable respectively. The model chosen is Hokuyo UST-10LX[5], which is depicted in Figure 4.3

[4] http://www.robotnik.eu/mobile-robots/rb-1-base/

[5] http://www.hokuyo-usa.com/products/scanning-laser-rangefinders/ust-10lx

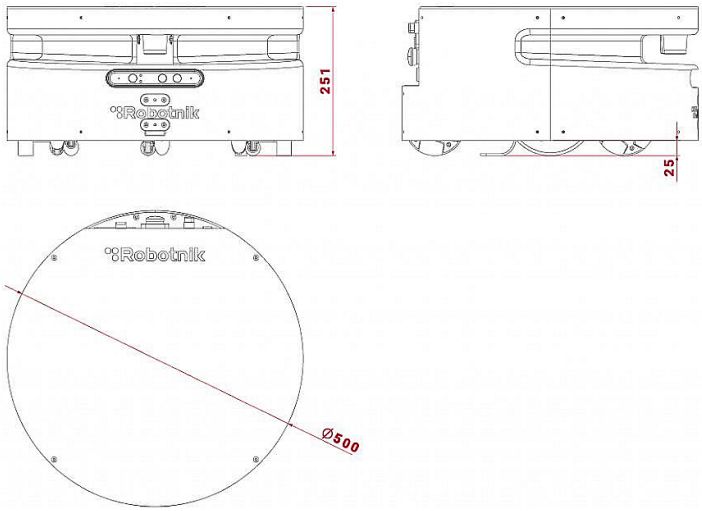
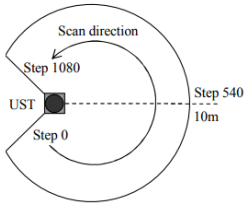


Figure 4.2: RB-1 mobile base schematics. The robotic subsystem is a customization of the commercial mobile base. Source: https://www.robotnik.es

with its main characteristics.



Detection angle	270°
Angular resolution	0.25°
Scan speed	25ms

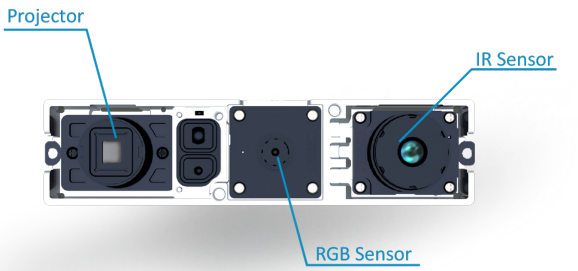
Figure 4.3: Laser Hokuyo UST-10LX, the one installed in the robotic subsystem. Source: http://www.hokuyo-usa.com/

The second essential sensors are colour and depth (RGB-D) cameras. Depth information is used by the navigation for obstacle avoidance. The

three-dimensional depth view of the robot vicinity serves the navigation in understanding where obstacles are present and where the path is clear for it to traverse. Given the intrinsic field of view (FOV) of one camera does not suffice to cover all the potential collisions with the robot's body, two RGB-D cameras are combined. The robot mounts the two depth cameras on the top of the identification subsystem. In this way, the combined FOV is maximized and, at the same time, dead angles are minimized. Additionally, the camera's RGB information leverages the detection of features on two-dimensional images. For instance, the detection of QRs codes placed for the identification of zones. Furthermore, camera views can be used for monitoring and teleoperation. Noteworthy, camera's streams allow supplementary applications of interest such as the three-dimensional mapping of the environment or objects detection. The camera model chosen is the Orbbec Astra Mini[6], a small form-factor and uncased camera that facilitates the integration on top of the identification subsystem body. It uses structured light technology, this is the projection of known light patterns and posterior interpretation of the reflection's deformation. The camera is interfaced and powered by means of a USB connector. Its main characteristics and a schematic are depicted in Figure 4.4.

The third sensor is the inertial measurement unit (IMU). It includes gyroscopes and accelerometers to measure linear and angular accelerations, which in turn are used to compute the robot's odometry. The odometry is an estimate of the robot's changes of position and orientation. The motor's encoders report the rotation of the two wheels and are also used for odometry computation. However, the rotation of the wheels does not necessarily imply robot's movements. For instance, wheel slipping, even unnoticeably, is common. Therefore, an accurate odometry is computed by fusing the motors encoders and IMU measurements. Still, the odometry is not precise enough for the computation of the robot's location by itself due to an incremental error accumulation. For this reason, a location algorithm is required. In brief, the robot's location is computed by applying a periodical correction to the measured odometry. The IMU installed

[6]https://orbbec3d.com/astra-mini/



Range	0.6m to 5.0m
Field of view	73D x 60H x 49.5V

Figure 4.4: Orbbec Astra Mini, the RGB camera model installed in the robotic subsystem. Source: https://orbbec3d.com/

in the robot is a Pixhawk Autopilot, used widely in drones and depicted in Figure 4.5 next to its main characteristics.

Figure 4.5: Pixhawk Autopilot, the gyroscope/accelerometer installed in the robotic subsystem. Source: https://pixhawk.org/

Hardware

Besides sensors, the mobile base includes other essential hardware.

Sensor data interfacing and fusion, actuators control, computation of algorithms, servicing an interaction interface and management of re-

sources at all levels are encapsulated in a computer embedded in the mobile base. The collection of all robot's logics are referred to as Brains, given their coexistence and interactions translate to actual robot reasoning and decisions. Both the abstraction and the hardware are referred to as Brains in this document.

Network connectivity is paramount for interacting with the robot. The Brains incorporates a WiFi client in order to connect the robot to an infrastructure network when available. Reasonably, any device in the same network could access the robot's services. Alternatively, the Brains includes a router that acts as an access point. In this manner, if an infrastructure network is not available or accessible, the connection with the robot can be established directly. Section 4.4 includes a detailed description of the interaction and connectivity with the robot.

The last essential piece of hardware is the battery. It enables an uninterrupted operation of up to 10 hours and a complete recharge in roughly 2 hours.

An interesting feature of the robotic subsystem construction is its modularity. It is composed of three independent modules: a structural part, which includes the motor-wheels; a box that encloses most of the electronics, but for the laser and the cameras, which are necessarily placed outside; and the battery. The electronics box and the battery can be easily removed and replaced from the structural part. Modularity is essential, once a robot is deployed, for the ease of servicing in case of failures.

4.2.3 Interconnection

The mobile base and the identification subsystems are designed thinking of a simple assembly and interconnection. The assembly of the two independent structures is achieved by just six screws. The interconnection requires establishing a communication. Communication is granted by two channels. First, an Ethernet link connects the Brains to the identification subsystem. Second, two USBs connect the depth cameras, installed on top of the identification subsystem, to the Brains. Last, the interconnection requires powering the identification subsystem, which is achieved by

a power link from the battery.

All the connectors are accessible through a small door on the bottom of the identification subsystem for an easy manual connection once the two subsystems are assembled.

A schematic of the assembled robot is shown in Figure 4.6.

4.3 Robot logic subsystem

The robot logic subsystem, or Brains, is the collection of software packages that command the robot behaviour. The packages extend the Robot Operating System (ROS) [7] framework. ROS is an open source framework for robotics development that implements a sort of blackboard system. Basically, ROS is organised around nodes, the actual entities that control, drive or compute things; topics, published and subscribed by nodes for information sharing; and services, the means for active interaction between nodes. ROS can be considered as the operating system of the Brains. On one hand, some of the packages that command the robot are community contributed and integrated or adapted to the specific robot characteristics. These include hardware drivers and a part of the navigation algorithms. On the other hand, the RFID-driven navigation control, a mission manager and a task manager are original contributions. In this section, an overview of the most relevant blocks and a detailed description of the contributed ones is given.

4.3.1 Navigation

The autonomous navigation relies on a map of the environment for planning the paths to a set of inventory goals. Consequently, a recognition round needs to be completed to create a map as a preparation for the autonomous navigation. Overall, the navigation can be considered a two-stage process. Recognition is a person-assisted procedure for the robot to gain knowledge about the environment, which includes recording a map.

[7] http://ros.org

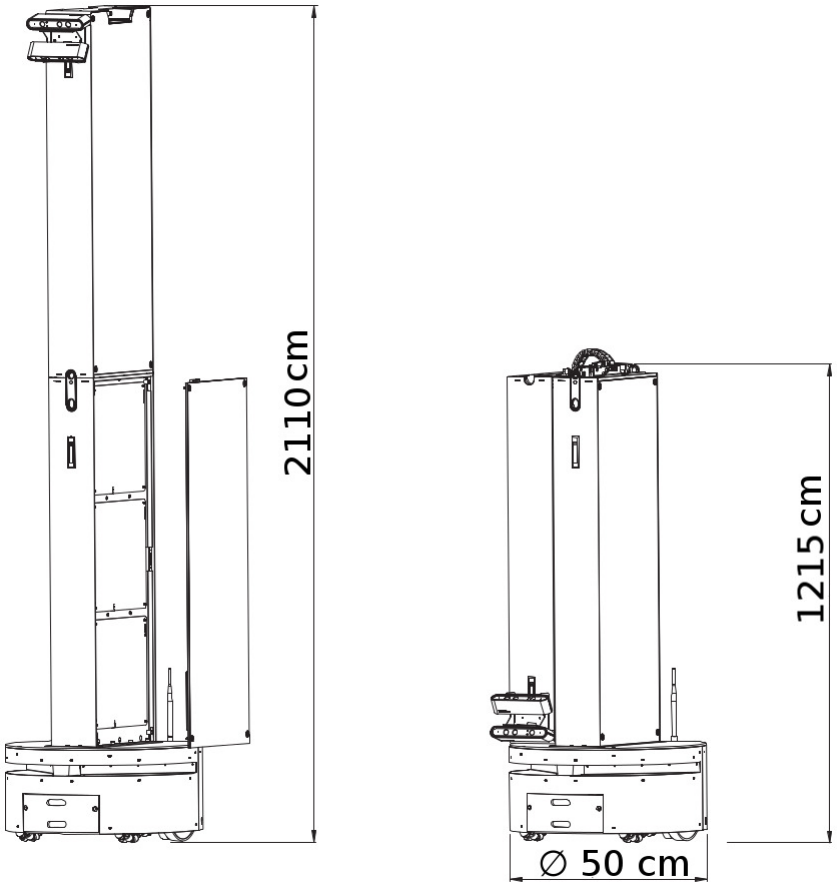


Figure 4.6: Schematics of the robot design. On the left, a view of the assembled robot, with the two RGB-D cameras on top of the RFID subsystem and a side cover removed to show a group of three antennas beneath. The bottom part is the mobile base. Note the laser’s 270°aperture and a front cover for a quick release of the battery. On the right, the payload is folded, a feature that was included for the ease of transportation.

Inventory is the stage at which stock information is gathered by the robot autonomously. The inventory stage can be triggered by an operator or without assistance.

The ROS navigation stack[8] is configured for the specific characteristics of the robot in order to achieve a safe and autonomous navigation.

Recognition

Recognition aims at collecting information of the environment in two different aspects. The first, recording a layout view, a two-dimensional map that can be used as a reference for the autonomous navigation. This is achieved feeding the laser range finder observations to a Simultaneous Localization and Mapping (SLAM) algorithm, which is implemented in ROS and called *gmapping*. The output is a view of the layout as a slice at the laser's height and looks like the map shown in Figure 4.12b. The second set of environmental data gathered during recognition are inventory waypoints. A series of coordinates are recorded by the robot which represent the specific locations that the robot should visit during inventorying.

Recognition is a guided procedure, an operator commands the robot using a control interface to move through an alleged inventory path, one that resembles the route followed when a handheld inventory is taken. The key idea is showing the robot all the locations that should be visited later at inventorying. As opposed to letting the robot just explore the space, which is time consuming in large open spaces, in this manner one can constrain the inventory target and, eventually, an inventory can be completed more efficiently. A specific purpose ROS package, named *goal_profiler*, is in charge of gathering inventory waypoints during the recognition procedure.

Inventory

Inventory is the stage at which actual stock-taking happens in a completely autonomous way. For that, the robot uses the previously recorded map and inventory waypoints. The autonomous navigation requires the robot being aware of its own location on the map, which is achieved by

[8] http://wiki.ros.org/navigation

applying Adaptive Monte Carlo Localization (AMCL) . Also, the navigation requires a means of computing the shortest passable path from the robot's current pose (position and orientation) to the next inventory waypoint. An implementation of the A* algorithm is used for global path planning. Importantly, in order to plan a path to reach a given waypoint, an understanding of passable and non-passable areas is needed. For that, a map that includes collision costs, called *global_costmap*, is initialized from the recorded map and continuously updated with live sensors data. In this way, any modifications to the environment are taken into account during navigation. At a lower level, once a global path is computed, proper commands need to be sent to the wheel motors to follow the track. Valid combinations of velocities that conform collision-free local trajectories are simulated by the *local_planner* and after transformed into robot movements. Figure 4.7 depicts the architecture of the navigation, which is part of the ROS distribution and parametrised for the robot specific characteristics.

At a higher level, inventory waypoints are managed by a specific purpose package named *goal_dispatcher*. It passes inventory waypoints to the navigation stack and monitors their progress. Remarkably, the navigation is based on inventory waypoints rather than on a predetermined path. When an inventory waypoint is passed to the navigation stack, the shortest possible path to reach that waypoint is computed. If the waypoint has become unreachable due to layout modifications, the waypoint is cancelled and the next waypoint dispatched. Basically, inventorying based on waypoint implies that the robot navigation can adapt to layout modifications. In this enterprise, monitoring the progress of waypoints is crucial to prevent an inefficient robot behaviour. Monitoring aims at identifying abnormal situations in attempting to reach a dispatched waypoint. On one hand, a timeout proportional to the distance and average robot speed is applied. This is justified by the fact that in presence of unexpected obstacles, the planned turnaround can become unacceptably long. On the other hand, given that the robot freely plans the path to an inventory waypoint, on its way it may pass by complementary waypoints. The monitoring flags unintentionally reached inventory waypoints in order not to dispatch

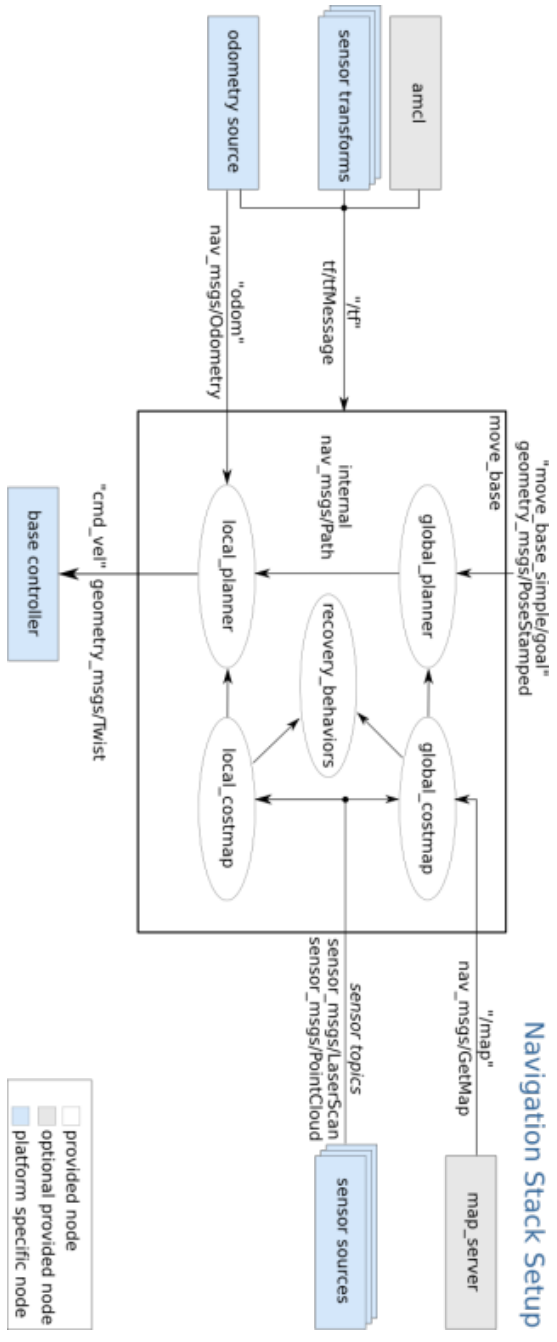


Figure 4.7: Architecture of ROS navigation stack, the logics used by the robot for autonomous navigation. Source: http://wiki.ros.org

them again.

Importantly, the behaviour of the navigation between inventory waypoints is determined by an event-driven control layer that takes as input the progress of RFID identifications. Such control is essential to guarantee accuracy and is detailed in Section 4.3.2.

A visualization of the navigation during an inventory is depicted in Figure 4.8.

4.3.2 RFID-driven navigation control

One of the main contributions of this work is the interaction between the behaviour of the navigation and RFID identifications. The premise is that for an accurate and time-efficient inventory, the robot must take into account the objects in scope from a specific location. Exploring possible extreme cases is illustrative. The first case would be that of a robot that navigates at its maximum speed constantly while taking inventory. Although the RFID protocol allows for simultaneous identifications, there is still a limitation on the maximum number of RFID tags read per second. Thus, in presence of a large amount of products, having time to read of all them is unlikely if passing by without stopping. The second case would be that of a robot that traverses aisles at a very slow pace to give enough time for identifications. In this case, the inventory would be accurate but not time-efficient.

Adjusting the robot behaviour to achieve an optimal compromise between accuracy and duration is inspired by RFID handheld readers. These include visual, auditory or haptic cues for the operator to know when all items at reach have been identified. The operator relies on such cues to decide when to move on to identify a next batch of products. In this way, the accuracy is not compromised and the operator does not waste unnecessary time in areas already queried. The robot applies an equivalent event-driven algorithm to command the navigation.

The RFID-driven navigation control algorithm takes as input the aggregated stream of identifications from the readers. The readers work in an asynchronous mode, putting in a queue the identification streams to

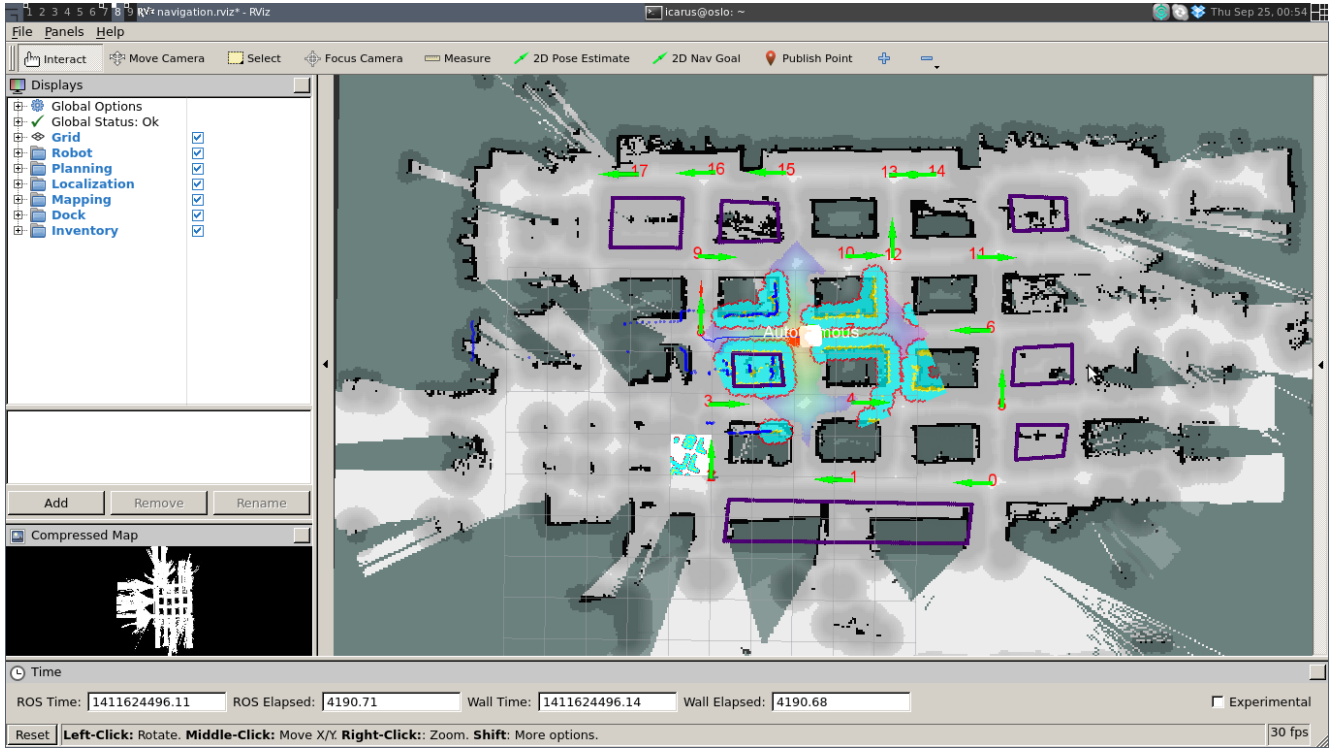


Figure 4.8: Visualization of the robot during an inventory round on RViz, the robot visualization tool provided with ROS. Several pieces of information are presented live and represent the robot’s knowledge of the environment. On one hand, the occupation map and waypoints, represented by arrows, which are static data recorded during recognition. On the other hand, live sensor data: in red laser scans; in light blue the collapsed camera view. Finally, robot computations: in yellow, the local path; in blue, the global path from an actual robot pose to the next navigation goal; in purple, the navigation costmap, which sets the costs for navigation based on sensor data.

be processed by the algorithm. As elements in the queue are processed, received product codes are added to a set used to track the history of identifications, the actual inventory. If the product code already exists in the inventory it means that the product was previously identified, thus it is ignored. If the product code does not exist in the inventory, it is considered a new product identification and is included in a time window to compute the throughput of new identifications, in items per second. As elements in the queue keep being processed, the time window rolls and the rate of new identifications updated.

The algorithm uses the rate of new identifications to trigger the transition between two states: *Twist* and *Journey*.

In *Twist* state, an interruption is sent to the underlying goal-based navigation in order to stop the robot for the identification of all products in scope. Instead of just remaining in place, the robot performs a periodical angular movement, the *twist*. The justification comes from the fact that even small movements help in overcoming the negative effects of multipath RFID interference and blind spots. The contribution of twisting to inventory accuracy is demonstrated in Chapter 5. In *Journey* state, the underlying goal-based navigation proceeds normally.

Transitions are triggered based on upper and lower thresholds applied to the rate of new identifications. Hence, state transitions happen in the likes of a hysteresis cycle, which prevents fast transitions and the consequent clutter in the robot's navigation. A proper adjustment of thresholds values is critical to achieve an efficient behaviour of the robot at inventorying, as is demonstrated in Chapter 5.

The RFID-driven navigation control is summarised in Algorithm 2.

Algorithm 2 Navigation control algorithm

1: $State \leftarrow Twist$
2: $I \leftarrow \emptyset$
3: $N \leftarrow \emptyset$
4: **procedure** CONTROLNAVIGATION(S_i)
5: $N_i \leftarrow S_i \setminus I$
6: $N \leftarrow N_i \cup N$
7: $N \leftarrow \{n \in N \mid (t_i - t_n) \leq T\}$
8: $R \leftarrow |N|/T$
9: **if** $R > th_{twist}$ **then** ▷ $th_{journey} < th_{twist}$
10: $State \leftarrow Twist$
11: **else if** $R < th_{journey}$ **then**
12: $State \leftarrow Journey$
13: **else**
14: $State \leftarrow State$
15: $I \leftarrow I \cup S_i$

I: inventory
N: new identifications
S_i: identifications at instant i
N_i: new identifications at instant i
T: time window width
R: rate of new identifications [items/s]

4.3.3 Mission manager

The robot is designed to perform stock-taking following a divide and conquer strategy. The justification comes from two different use cases. The first, a retail store that involves a large amount of square meters that cannot be inventoried in a single inventory run. The second, running the robot as a service, which implies inventorying alternatively different spaces. The divide and conquer strategy allows for flexibility, scalability, reliability and efficiency.

In terms of flexibility, it allows the possibility of inventorying a se-

lected combination of zones under demand. Also, if a significant layout change is detected in one of the zones, the recognition should be rerun only at that specific zone. Regarding scalability, it is possible to add and reconfigure zones as needed. In terms of efficiency, navigation in smaller zones is less demanding in terms of computation and also quicker.

The divide and conquer strategy consists in dividing a store layout into zones. Zones are organised sequentially, which means that the starting point of one zone is the ending point of the previous one. In this way, all the zones are linked. The zones starting and ending points are marked with QR codes for their automatic identification by the robot. At the same time, QR codes serve as visual cues for associates to identify the zones and launch the recognition stage. Zones configuration and installation of QR codes is a preparation step that can be completed by the final user. Figure 4.9 shows an example of an actual 7,500 m^2 store divided into 22 zones that was inventoried by the robot.

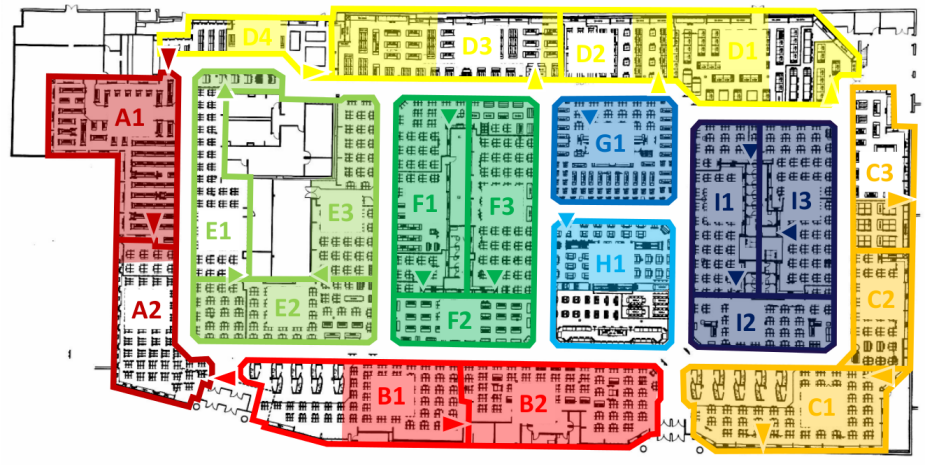


Figure 4.9: Store layout divided in 22 zones, following the divide and conquer strategy implemented by the robot. The triangles are the cues of the QR codes locations. The store area is 7,500 m^2.

The middleware that manages the operation with zones is called mis-

sion manager. A mission is defined as a combination of zones to inventory. Any combination is allowed as long as the robot is placed at the selected starting zone. A mission always follows an ordered sequence of zones, inventorying those that are selected and skipping those that are not. Skipping a zone implies that the robot, instead of attempting to reach all the inventory waypoints, moves directly to the zone's end point. The mission manager implements the logic to initialize and monitor the progress of the robot through zones. In brief, it serves maps and inventory waypoints at the appropriate moments for a mission to be completed.

4.3.4 Task manager

The task manager is the block that translates high-level user orders to lower-level robot control commands. It provides the necessary level of abstraction for a user-friendly interface. Task manager executes selected task actions using the ROS actionlib stack [9] and monitors their progress in order to trigger consequent actions. Also, it keeps the state of the robot to prevent the interference of tasks. For instance, while the robot is inventorying, it is not possible to trigger a second inventory. In brief, task manager works as a simple finite state machine that launches the proper nodes and triggers the proper actions at the proper moments for a high-level task to be completed. Furthermore, it centralises relevant information from all the robot nodes. On one hand, such information is used to take action if needed. For instance, if the battery is too low, an ongoing inventory mission is cancelled. On the other hand, the task manager passes information to the interface for its presentation to the user, such as the status and progress of a mission as can be seen in Fig. 4.13c.

[9] http://wiki.ros.org/actionlib

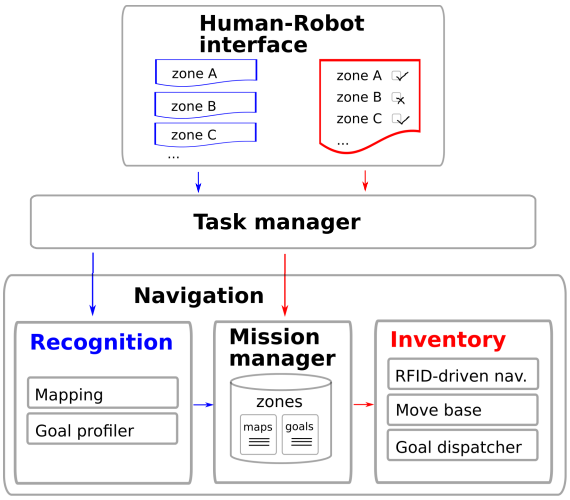


Figure 4.10: The task manager takes high level task parameters from the Human-Robot interface and translates them to robot orders by following the proper sequence of states that trigger all the nodes needed during a mission. The status of the robot is monitored by the task manager and implements event-based state transitions. The mission manager connects the recognition and the inventory. Namely, it manages the sequence of zones selected during an inventory mission. Mainly, it loads the map and goals of each zone at the proper zone transitions.

4.4 Human-Robot interface

The Human-Robot interface (HRI) is a web-based interface supported by any browser-enabled device. Actually, the accessory provided for interaction with the robot is a commercial mobile phone. The interface makes use of the rosbridge procotol[10] and additional specific RESTful services for interaction.

The interface is served by the robot through two separate wireless links: an infrastructure and an ad-hoc. The first allows the access to the robot from any remote location as long as the robot is granted global connectivity. When the former is not possible the robot can be connected ad-hoc. In this way, the HRI is infrastructure-agnostic and accessibility is

[10] http://wiki.ros.org/rosbridge_suite

guaranteed for operation in any situation. At the same time, if an infrastructure network is available, the robot can be accessed for monitoring and teleoperation remotely.

The interface main menu provides access to five areas for interaction:

- Inventory and Recognition, which facilitate the two main procedures for operation, described earlier in Section 4.3;
- Reports, which provides historical information of all the tasks completed by the robot;
- Guidelines, intended to aid the user at robot's operation;
- Administration, which includes advanced configurations and operation procedures.

Figure 4.11 shows the interface's home page and a description of status icons and buttons.

4.4.1 Recognition and inventory

Recognition and inventory are the two main procedures for robot operation. They share common initial steps and both benefit from QR codes for the automatic identification of zones. The use of QR codes is key for the divide and conquer strategy at inventorying large spaces (Section 4.3). A QR code is placed at the starting point of every zone. QR codes serve as a cue both for the operator and the robot to identify zones. Thanks to the QR codes an operator knows where the robot should be placed to start. After, the robot gains knowledge of its initial position by automatically detecting the QR code. Recognition and Inventory steps are listed next and the corresponding interface views are referenced within brackets.

Recognition

1. Place the robot in front of a zone's QR identifier.
2. Select Recognition in the Main Menu.

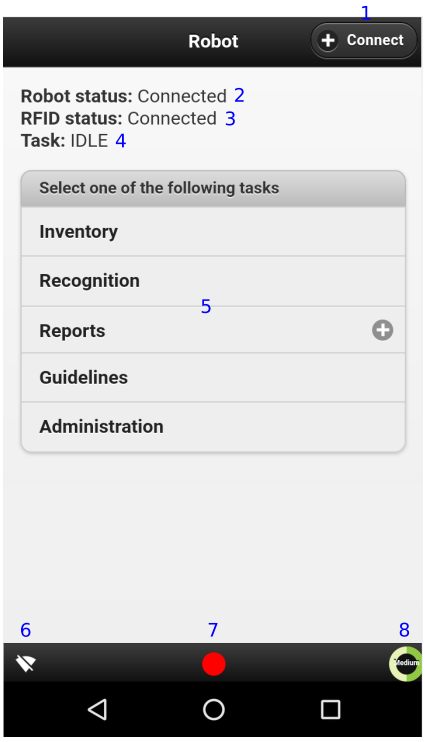
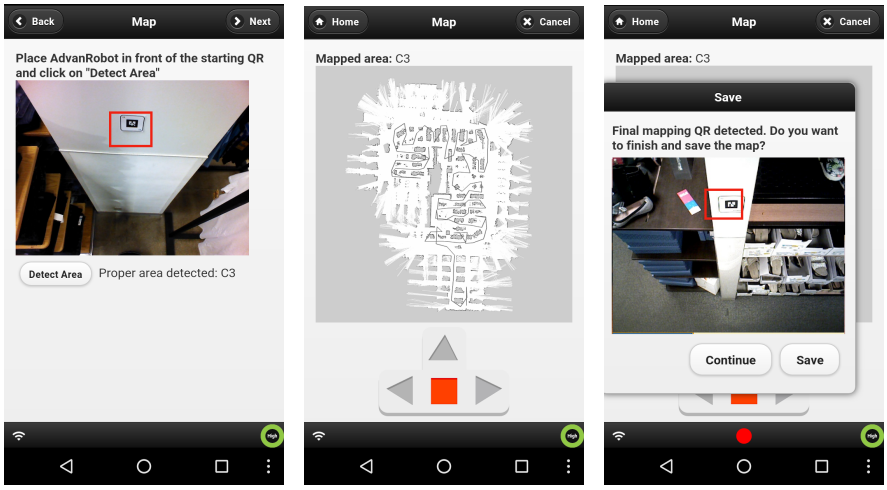


Figure 4.11: Home page of the interface that enables the interaction with the robot.

3. Adjust the robot’s position for the QR to be inside the red mark (Fig. 6.4b).
4. Press Detect Zone (Fig. 6.4b).
5. Verify that the detected zone coincides with the intended zone (Fig. 6.4b).
6. Verify that the emergency stop button is not pressed.
7. Press Start.
8. Command the robot using the pad. The robot should be guided resembling the path followed at manual stock-taking. The interface shows an updated view of the recorded map and interest points connected by line segments (Fig. 4.12b).
9. Bring the robot in front of the next zone’s starting QR.

10. Accept finishing or continue the recognition (Fig. 5.2c).



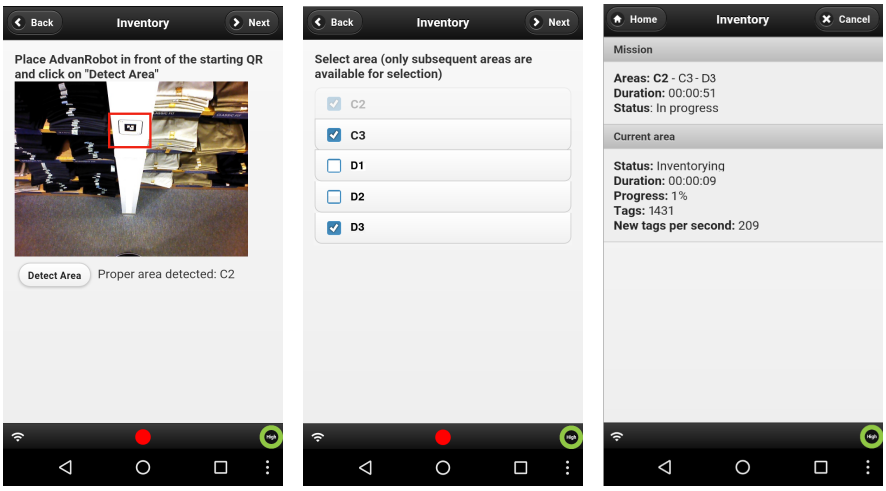
(a) **(b)** **(c)**

Figure 4.12: Interface views at running a recognition. (a) Automatic detection of the QR that identifies a zone at its starting point. (b) During the recognition a control pad allows commanding the robot. A view of the recorded map with interest points linked by straight lines is updated continuously. In this way, the operator can consult the progress of the recognition and oversights can be prevented. (c) Automatic detection of the QR code that identifies the immediate consecutive zone, which marks the end of the current zone. In this manner, transitions between zones are guaranteed. At this position the recognition can be concluded or continued. In any case, a recognition is necessarily finished in front of the QR that marks the start of the next zone.

Inventory

1. Place the robot in front of a zone’s QR identifier.
2. Select Inventory in the Main Menu.
3. Adjust the robot’s position for the QR to be inside the red mark (Fig. 4.13a).
4. Press Detect Zone (Fig. 4.13a).

5. Verify that the detected zone coincides with the intended zone (Fig. 4.13a).
6. Select the zones for the current inventory mission. Unselected zones will be transitioned (Fig. 4.13b).
7. Verify that the emergency stop button is not pressed.
8. Press Start. The robot starts taking inventories autonomously.
9. The interface informs about the progress of the ongoing inventory mission (Fig. 4.13c).

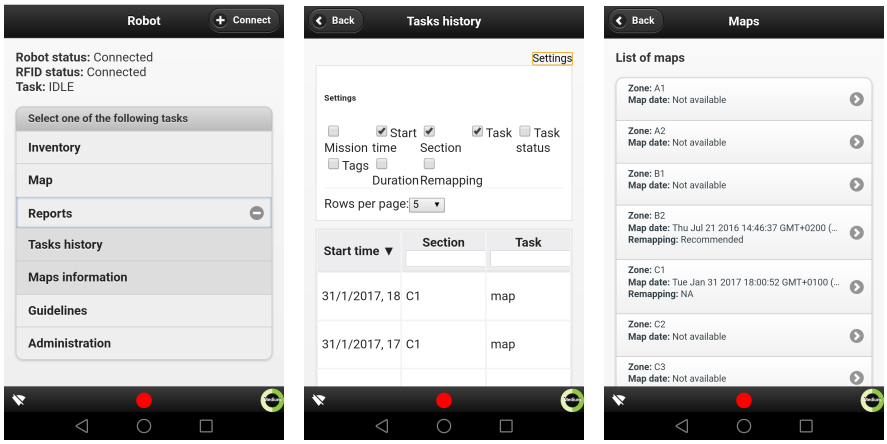


(a) **(b)** **(c)**

Figure 4.13: Interface views at running an inventory. (a) Automatic detection of the QR that identifies a zone at its starting point. (b) Selection of zones to be inventoried. Unselected zones are transitioned as opposed to inventoried. (c) Inventory progress view. It is divided in two blocks. Mission shows global information. It lists the zones configured in the current mission (C2-C3), total duration and status. The second block, Current area, shows information about the ongoing inventory, which corresponds to a single zone.

4.4.2 Reports

Reports consists of two views of information of interest to the final user. The first view, a log of tasks undertaken by the robot along with relevant information fields, which can be opted in or out (Fig. 4.14b). The second view, the status and visualization of recorded maps (Fig. 4.14c). This is important to know if the recognition has been accomplished in all the zones and whether it needs to be rerun after significant layout modifications, which is automatically informed.



(a) **(b)** **(c)**

Figure 4.14: Interface views under the Reports menu. (a) Reports is divided in two submenus: Tasks history and Maps information. (b) Tasks history allows the consultation of every task undertaken by the robot. (c) Maps information lists configured zones and whether the recognition has been completed (Map date). As well, it suggests rerunning the recognition when significant layout changes are detected (Remapping).

4.4.3 Administration

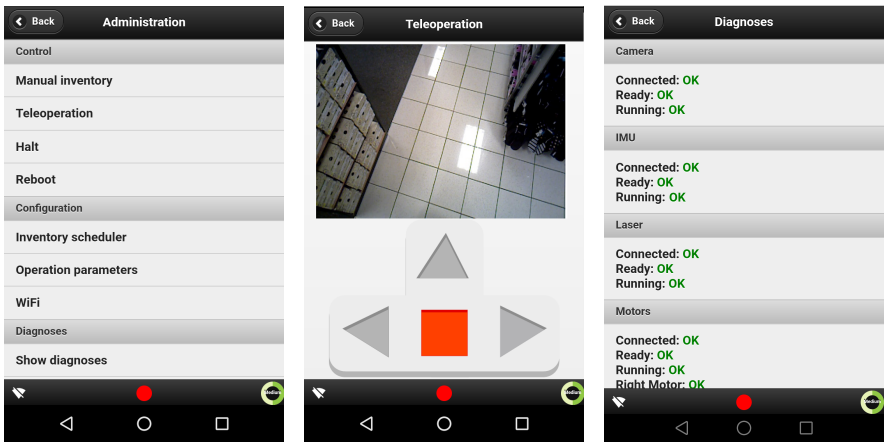
The Administration area includes a set of low-level controls and features, which are restricted to operators with advanced knowledge and privileges. The Administration toolkit consists of the following:

- **Manual inventory** An operator can run a non-autonomous inventory during which the robot is commanded from the interface. The control view is analogous to Teleoperation (Fig. 4.15b). Initial steps are equivalent to those of the Inventory procedure, requiring a QR code for the identification of the zone.
- **Teleoperation** It allows viewing the live camera stream and controlling remotely the robot.
- **Halt** Switches off the Brain.
- **Reboot** Reboots the Brain.
- **Inventory scheduler** Provides a means to configure the robot on a weekly schedule of zones. Scheduling zones is useful in large stores where inventories cannot be completed in a single battery load or working shift
- **Operation parameters** Low level configuration parameters.
- **WiFi** Configuration for the connection to a WiFi network.
- **Show diagnoses** Runs a diagnostics tool that checks the status of hardware for first-level troubleshooting 4.15c.

4.5 Robot design validation

The robot design was validated by 11 untrained users that were asked to complete, first, the assembly of the robot parts and, after, recognition and inventory procedures. In order to accomplish the tasks, participants were supplied with the robot's manual and given no additional explanations. Upon completion, participants were asked to fill in a questionnaire for the assessment of the experience. Among participants, there were actual potential customers that undertook demos with the robot.

Table 4.2 summarises the responses to the questionnaire related to the assembly. While assembly and disassembly are tasks that can be thought to be performed infrequently, this might not be the case always. Thinking of the robot as a service, as opposed to a resident device, its mobility from store to store is critical. Despite the tasks involve only the connection



(a) **(b)** **(c)**

Figure 4.15: (a) Robot’s control interface Administration menu. (b) Teleoperation allows the remote commanding of the robot and shows a live stream of the camera as reference. (c) Diagnostics view is intended for troubleshooting. It lists the status of all relevant hardware parts.

of four cables and fixing six screws, it is still not deemed as extremely simple. The main reason, pointed repeatedly in the additional comments, is the weight of the parts, most noticeably the identification subsystem, which needs to be lifted. In this regard, next design iterations should consider making the parts lighter for a better manageability.

Statistics of the responses related to the operation of the robot, which includes inventory and recognition procedures, are shown in Table 4.3. One must note that such procedures involve using the robot’s interface. Overall, responses are satisfactory. Given that participants were simply given a manual, the conclusion is that both procedures are simple and clear enough. The focus for improvement is on the interface, which is deemed as helpful but, as additional comments stress out, it could include more guidance. This was expected since the interface was developed as a minimum viable solution for operation and can be enriched with visual and textual clues and its design enhanced. Given the robot is meant to be operated by unskilled users, this point should be addressed in next

Table 4.2: Results of the questionnaire for the validation of robot assembly. The statements were valued between 1 (totally disagree) and 5 (totally agree).

Statement	**Answer average**	**Answer standard deviation**
It’s simple to assemble/disassemble	3.3	1.0
It’s easy to learn to assemble/disassemble	4.1	0.7
The effort to assemble/disassemble is acceptable	3.7	0.9
The time to complete the task is acceptable	3.9	0.9

iterations.

Table 4.3: Results of the questionnaire for the validation of robot operation. The statements were valued between 1 (totally disagree) and 5 (totally agree).

Statement	**Answer average**	**Answer standard deviation**
It’s simple to use	4.0	0.9
It’s easy to learn to use	4.2	0.8
The interface is intuitive and guides you through the process	3.2	0.8
The time to complete the task is acceptable	4.1	0.7

Contrastingly, operation procedures are considered simpler than assembly and disassembly. This is a good point since operation procedures

are expected to be run more frequently.

Chapter 5

INVENTORY

The automation of inventory taking is one of the ultimate goals of this work. The assessment of inventory accuracy and its duration is addressed thoroughly, focusing on the robot's navigation behaviour adjustment for the optimisation of results. An important contribution is the definition of a methodology for the assessment, given the lack of previous works addressing this topic. This includes insight into the computation of a baseline and the definition of specific measures. Also, the verification of the robot as a device for automatic inventory and the setup of a verification environment are addressed. Ultimately, experimentation at a number of retail stores and a thorough comparison with the procedure of taking inventories with RFID handhelds is presented.

5.1 Methodology

5.1.1 Accuracy computation

The computation of inventory accuracy presents a notable challenge. Fundamentally, the lack of a reliable baseline for the assessment given that usual inventory records and counting methods are less accurate than the robot inventory to be evaluated. Furthermore, one must note that the targeted environments involve tens or hundreds of thousands of items where

manual counting requires a prohibitive amount of work.

Generally, libraries and stores keep a record of available items by updating the count as they are consumed and replenished. This is commonly known as a perpetual inventory. Unfortunately, perpetual inventories diverge significantly from reality due to wrongly recorded transactions, theft and misplacement.

In addition, periodic corrections are applied following manual audits. Such audits are indispensable and known as fiscal inventorying, given enterprises should include accurately assets in their balances for tax reporting. To some extent, manual counts are undertaken assisted by automatic identification technologies such as barcodes or RFID. In any case, such procedures are not frequent and involve the action of persons with consequent unintended errors.

Hence, perpetual inventories, corrected or not, are not a reliable picture of what there actually is in a store, although they can be used as a support.

While the ideal baseline for the assessment of robot inventories would be a manual count, its closest realistic expression is taking inventories with a handheld RFID device. Interestingly, this is the most widespread way of taking RFID inventories nowadays, and so a reasonable reference to consider and challenge.

Therefore, two sources of inventory information are considered: perpetual inventory records; and RFID handheld inventories. Additionally, a database of product references is essential in order to filter out products not in a mission's scope. In explanation, at inventorying a selected zone, products from neighbouring areas are usually identified. Given that the robot's reading reach is significantly longer than a handheld, the robot is able to gather more product codes from distant zones. Hence, if those are not filtered out, the accuracy figure would be distorted in favour of the robot.

Three main inventory sets are considered in the computation of baseline and accuracy in this work. The inventories enclose data of different nature. On one hand, inventories taken with RFID devices provide universal unique item identifiers, expressed by the Electronic Product Code

(EPC). On the other hand, the perpetual inventory data generally consists of product type identifiers, Stock Keeping Units (SKU), which are not unique universally. An EPC encloses the item's corresponding SKU code, which makes the comparison of the two different inventory types possible.

- $P_{1..n} = \{(sku_1, \sharp sku_1), ..., (sku_O, \sharp sku_O)\}$ represents the set of items listed in a perpetual inventory, expressed as pairs of product reference code (SKU) and quantity. The perpetual inventory includes the items within n zones;
- $R_{zi} = \{epc_1, ..., epc_U\}$ represents the set of items identified at an ith round of robot inventorying an arbitrary zone z. By decoding the EPCs, it can be equivalently expressed as $R_{zi} = \{(sku_1, \sharp sku_1), ..., (sku_M, \sharp sku_M)\}$
- $H_{zi} = \{epc_1, ..., epc_V\}$ represents the set of items identified at an ith round of manual RFID handheld inventorying an arbitrary zone z. By decoding the EPCs, it can be equivalently expressed as $H_{zi} = \{(sku_1, \sharp sku_1), ..., (sku_N, \sharp sku_N)\}$

A complementary data set used in the computation of baseline and accuracy is the set of product types within a specific zone where the inventory is taken, expressed as $Z = \{Z_1, ..., Z_n\}$, being $Z_z = \{sku_1, ..., sku_L\}$ the set of product types within zone z. This is the data needed to group inventory data by zones and enables filtering out the items detected that do not belong to the zone in scope.

Figure 5.1 depicts the information sets considered and their overlaps, which are the focus for accuracy computation.

The availability of data varies depending on the scenario. Accordingly, the baseline, thus accuracy, will rely upon varying combinations of data. Next, three possible accuracy figures are introduced: raw accuracy; targeted accuracy; and verified accuracy. In all of them the baseline is a fusion of all the information available, including multiple handheld and robot inventory rounds in the same batch. Table 5.1 summarises the accuracy measures and the required sets of data for their computation.

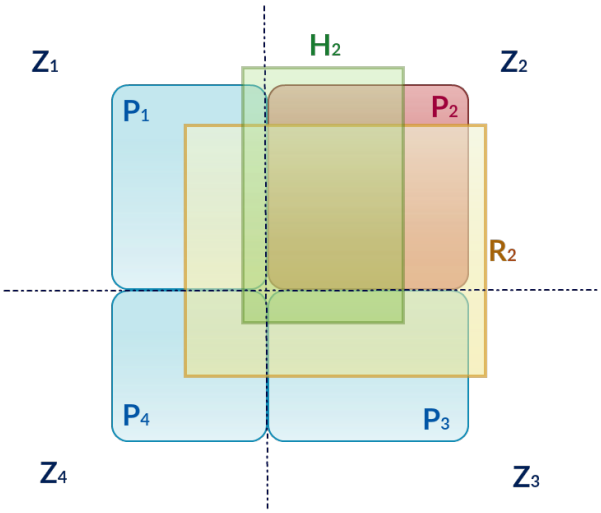


Figure 5.1: Data subsets considered at accuracy computation. Four zones are depicted, expressed by their corresponding product types: Z_1, Z_2, Z_3 and Z_4. The inventoried zone in this case is Z_2. The set of handheld detections is H_2, the set of robot detections is R_2 and the perpetual inventory is P_2. Usually, both the robot and the handheld identify items that belong to zones adjacent to the zone in scope. In the case of the robot, this is usually more significant.

Onwards, for notation simplicity, it is assumed that inventories are taken for an arbitrary zth zone although not explicitly expressed.

Raw accuracy

Raw accuracy is the measure used when only handheld and robot counts are available. Raw accuracy relies on the premise that, if the area covered by both devices is the same, everything that is identified by the handheld should then also be identified by the robot. In this case, the baseline is the aggregation of one to multiple handheld and robot inventory measurements during a period where the stock is known to remain unchanged. Equation 5.1 describes the computation of the raw baseline using N handheld measurements and M robot measurements. Equation 5.2 describes the robot raw accuracy for an i_{th} robot round.

	R_{zi}	H_{zi}	Z	$P_{1..n}$
Raw accuracy	x	x		
Filtered accuracy	x	x	x	
Verified accuracy	x	x	x	x

Table 5.1: Summary of accuracy measures and the data sets required to compute them.

$$B_{raw} \equiv \bigcup_{i=1}^{N} H_i \cup \bigcup_{i=1}^{M} R_i \tag{5.1}$$

$$Acc_{raw}(R_i) \equiv \frac{|R_i \cap B_{raw}|}{|B_{raw}|} \tag{5.2}$$

$$|B_{raw}| \equiv \text{Cardinality of } B_{raw}$$

Given the robot identification reach exceeds considerably that of the handheld, the equivalent measure to assess handheld accuracy is underrated. For this reason, it is discarded as a representative measure. Given the raw accuracy is not useful for accuracy comparison, it is not considered onwards in this thesis.

Filtered accuracy

The filtered accuracy is the measure used when information about product types in scope is available. Products identified that do not belong to the inventory target zone are filtered out. This allows for a fair comparison between the robot and handheld accuracy, given identifications are confined to the zone in scope.

Equation 5.3 describes the filtered baseline computation.

$$B_{filtered} \equiv \left(\bigcup_{i=1}^{N} H_i \cup \bigcup_{i=1}^{M} R_i\right) \subset Z_n = B_{raw} \subset Z_n \tag{5.3}$$

In order to achieve a baseline that is faithful to the actual items count, handheld and robot inventory iterations, N and M, have to be increased until a residual baseline increment is reached.

Equation 5.4 describes the accuracy computation for an i_{th} round of an inventory I taken either by the robot or the handheld.

$$Acc_{filtered}(I_i) \equiv \frac{|I_i \cap B_{filtered}|}{|B_{filtered}|} \qquad (5.4)$$

$$I_i \in \{H_i, R_i\}$$

Importantly, neither the filtered nor the raw accuracy take into account items in scope not identified by any means. It is an insightful comparative measure, although it could become misleading if the quality of inventories is low from all sources at the same time. Interestingly, if just one of the inventory sets is erratic, the filtered accuracy will disclose it.

Verified accuracy

The verified accuracy extends the aforementioned measures by using the perpetual inventory record as a complementary reference. Albeit the perpetual inventory record itself could be used directly as the baseline, given it is known to diverge from reality, it is not considered as is. Instead, a thorough procedure that takes into account all the inventory data sets available is followed aiming at measuring the most accurate baseline possible.

Essentially, the procedure consists in analysing the discrepancies between the filtered baseline and the perpetual inventory. Products that show inconsistent item counts between the two records are searched for manually to confirm whether the discrepancy is due to a wrong baseline count or due to a wrong perpetual inventory count. After the manual counting, the actual count is used to correct the baseline and the perpetual inventory. In the end, both records converge to a faithful count of items, what we call the verified baseline.

The procedure starts by computing a set of alleged false negative detections $\widehat{M}$ by subtracting the positive detections in all rounds, namely $B_{filtered}$, to the items listed in the perpetual inventory P_z.

$$\begin{gathered}\widehat{M} \equiv P_z \setminus B_{filtered} \qquad\qquad P_z = P_{1..n} \subset Z_n \\ P_z \setminus B_{filtered} \equiv \{sku \in P_z | sku \notin B_{filtered}\}\end{gathered} \tag{5.5}$$

$\widehat{M}$ is an estimation of the items missed by the robot and the handheld. Note that RFID never outputs false positive detections, which means that every item identified is indisputably present. On the contrary, items listed in the perpetual inventory are not necessarily in scope.

Next, items contained in $\widehat{M}$ are manually searched for. If an item is found, it is removed from the shelf and its identification with an RFID handheld is attempted. This is done to discard a malfunctioning RFID label. Occasionally, RFID labels are damaged or wrongly coded. Items that can be identified with an alternative RFID device are added to the set M, which contains the actual missed items after verification.

$$M \equiv \{\widehat{m_i} \in \widehat{M} \mid \widehat{m_i} \quad \text{is found and detectable}\} \tag{5.6}$$

To conclude, the verified baseline is computed as the union of true positive detections, or filtered baseline, and actual misses.

$$B_{verified} \equiv B_{filtered} \cup M \tag{5.7}$$

Finally, the verified accuracy can be computed as described in Equation 5.8.

$$\begin{gathered}Acc_{verified}(I_i) \equiv \frac{|I_i \cap B_{verified}|}{|B_{verified}|} \\ I_i \in \{H_i, R_i\}\end{gathered} \tag{5.8}$$

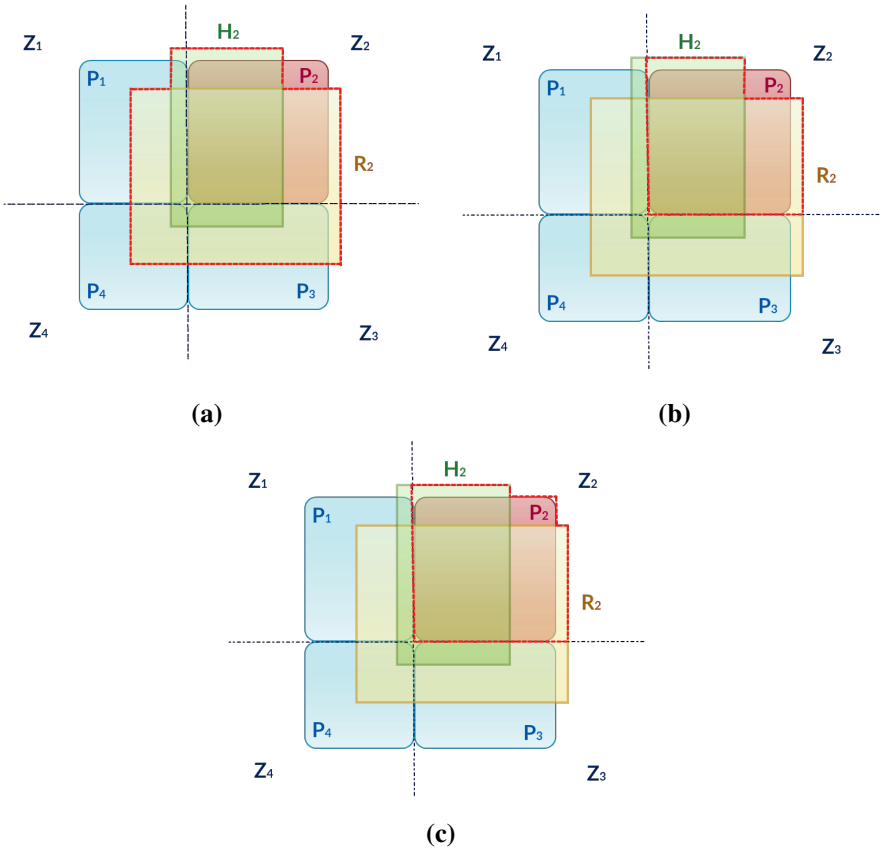


Figure 5.2: Visualisation of the different baselines - delimited with a dashed red line - used in accuracy computation. The use of one or other baseline measure depends on data availability. (a) Raw baseline (B_{raw}). (b) Filtered baseline ($B_{filtered}$). (c) Verified baseline ($B_{verified}$).

5.1.2 Definitions

The definition of specific measures is motivated by the fact that no former works have addressed a similar topic. To the best of my knowledge, no measures have been previously defined for the comparative evaluation of inventorying in different environments and using diverse methods. A first group of measures are related to the characteristics of a layout, which

influence the performance at inventorying: aisles length; intricacy; and items density. A second group express the actual performance: effective speed; and effective read rate.

Aisles length

We call aisles length, L_{aisles}, the total length of aisles encompassed in a given layout, in meters.

$$L_{aisles} \equiv \sum_{i} \ell_i \qquad [m] \tag{5.9}$$

In contrast with the area, the aisles length gives an idea of the extent of the inventory task, since it gives the actual distance a robot or person should eventually travel to identify all the items.

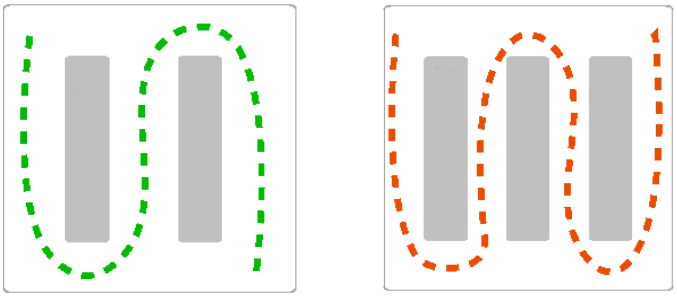
Noteworthy, although the robot is able to identify items across shelves, in the presence of specific materials, such as metals or liquids, the RFID signal can be blocked. Therefore, rather than relying on its reading reach for planning the navigation, the robot, equivalently to a manual inventory, is configured to visit all the traversable aisles. In this manner, unpredictable signal blockages are minimized and the accuracy is not compromised.

Intricacy

The speed of a robot navigating a space depends on its proximity to obstacles and the type and amount of turns involved. When obstacles are close by, the cruise speed is reduced for caution and at turns the robot speed is adapted. Therefore, a complete navigation in wide and straight aisles is faster than in narrow aisles with many turns. Consequently, the duration of inventories will be dependent on the characteristics of the layout.

Intricacy is computed as the aisles distance that correspond to each unit of area, expressed in Equation 5.10.

$$\text{Intricacy} \equiv \frac{L}{A} \qquad [m/m^2] \tag{5.10}$$



(a) Lower intricacy **(b)** Higher intricacy

Figure 5.3: Within the same area, layout (b) includes more aisles length than layout (a). Therefore, the degree of intricacy in layout (b) is higher than in layout (a). Note that a complete navigation of layout (b) involves three turns in narrower aisles in contrast with the two turns and wider aisles in layout (a).

Intricacy can be thought of as the distance that should be travelled to scan a square meter. Indeed, it gives an idea of the intricacy of the layout, since the more meters within a square meter, the more turns expected. Further, it is a comparative measure of the width of the aisles when shelves at the compared layouts are similar in size. Figure 5.3 is a graphical representation of two simple layouts with different levels of intricacy.

Items density

At stocktaking, a high amount of items per unit of area can be a notable challenge. A usual consequence of a significant amount of references in a confined space is their placement in a very packed manner, which means RFID labels can be occluded or interfere with each other. In addition, the more items the RFID signal has to traverse, the more it is weakened. Items density is the amount of items per area unit, expressed in $items/m_2$. It is computed from the baseline measure at use and expressed in Equation 5.11.

$$\sigma_{items} \equiv \frac{|B|}{A} \qquad [items/m_2]$$
$$B \in \{B_{filtered}, B_{verified}\} \tag{5.11}$$
$$|B| \equiv \text{Cardinality of } B$$

Additionally, the items lineal density, the amount of items per aisles length unit, is described by Equation 5.12.

$$\lambda_{items} \equiv \frac{|B|}{L_{aisles}} \qquad [items/m]$$
$$B \in \{B_{filtered}, B_{verified}\} \tag{5.12}$$
$$|B| \equiv \text{Cardinality of } B$$

Combining Equations 5.11 and 5.12, Intricacy can be expressed in terms of items density and items lineal density as

$$\text{Intricacy} \equiv \frac{\sigma_{items}}{\lambda_{items}} \qquad [m/m^2] \tag{5.13}$$

Effective speed

A robot that navigates in a cluttered and dynamic environment periodically faces situations in which it needs to reroute and walk around. For instance, due to unexpected obstacles or narrow thus impassable aisles. On the contrary, a person can easily manoeuvre in such situations. As a result, robot travelled distances are generally longer than the optimal path, the total aisles length, while a person's journey is equivalent.

The effective speed is computed disregarding the excess of distance travelled and focuses on the aisles length. In this manner, the effective

speed expresses the actual time to inventory a meter of the layout. Furthermore, an inventory round that does not deliver the maximum accuracy has to be penalised thus a correction factor η is applied. The reason is that if the accuracy of the round is not 100%, the time taken by the inventory should have been longer. Additionally, for the measure to be comparable across different inventory rounds, it needs to be normalised. Equation 5.18 reveals the computation of the effective speed.

$$v_{eff} \equiv \frac{L}{\Delta t} \cdot \eta(Acc) \qquad [m/s] \tag{5.14}$$

The normalization factor η is defined as a function of the accuracy. Looking at a typical plot of accuracy over time (Figure 5.4) one can see that the contribution of time to accuracy is not linear. The typical behaviour is an irregular growth of the accuracy during the initial time period and a final very slow growth towards the 99% and higher accuracies. Indeed, the final slow growth is contributed by the difficult identifications, which are the critical to achieve an accuracy higher than 99%.

The evolution of accuracy over time resembles a logarithmic growth. Let t_{100} be the duration of an inventory that delivers a 100% accuracy and Acc the accuracy. The contribution of time to the growth of accuracy can be approximated as a logarithmic function

$$Acc(t) \approx \log_2 (\frac{t}{t_{100}} + 1) \tag{5.15}$$

From equation 5.15 an expression of time is extended

$$\frac{t}{t_{100}} \approx 2^{Acc(t)} - 1 \tag{5.16}$$

The function used as the correction factor in the computation of the effective speed extends from equation 5.16. Intuitively, it expresses the duration taken by an inventory compared to the duration needed for a perfect accuracy. The function meets the requirements of the correction factor. A slow initial growth and a steeper growth towards the perfect

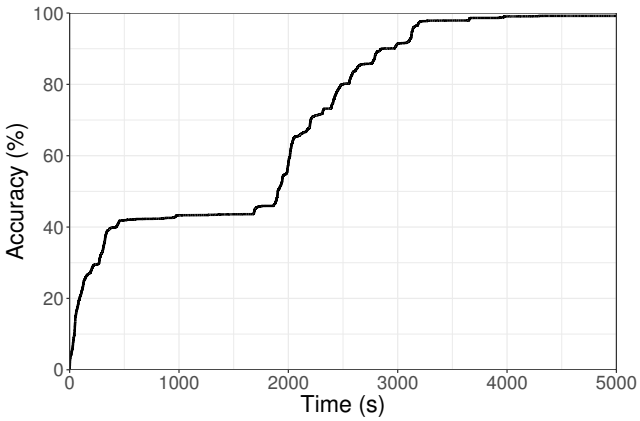


Figure 5.4: The evolution of accuracy over time in a real-world scenario. A fast growth is followed by a slow growth. Around time 2000 s the discovery of a new crowded area restarts the typical growth sequence. Effort does not contribute linearly to accuracy. Note the two quasi-flat regions, where a prolonged time period is needed to gain a small share of accuracy. The explanation is that easy identifications happen in bulks while difficult ones need a continued effort.

accuracy. In this way, low accuracies are strongly penalised while an accuracy close to 100% gets a much smaller penalisation. A 100% accuracy gets no penalisation at all.

$$\eta(Acc) = \frac{\Delta t}{t_{100}} = 2^{Acc(\Delta t)} - 1 \tag{5.17}$$

To sum up, the effective speed represents the actual speed at yielding a 100% accuracy and, combining equations 5.18 and 5.17, can be expressed as

$$v_{eff} \equiv \frac{L}{\Delta t} \cdot (2^{Acc(\Delta t)} - 1) \qquad [m/s] \tag{5.18}$$

Effective read speed

The effective read speed measures the throughput of new identifications. That is, the new identifications per time unit that are actually registered

during an inventory, expressed in $items/s$. It depends on the navigation duration and the density of items. However, it is an insightful comparative measure of a device's actual reading capacity in a given environment. It is not to be confused with the read rate, which is usually defined as the number of identifications, not necessarily new, that can processed in a given time period by an RFID reader. Equation 5.19 describes the computation of the effective read speed for an inventory I_i.

$$\begin{aligned} rs_{eff} &\equiv \frac{|I_i|}{\Delta t} \qquad [items/s] \\ I_i &\in \{R_i, H_i\} \\ |I_i| &\equiv \text{Cardinality of } I_i \end{aligned} \tag{5.19}$$

5.2 Verification and optimisation

The verification and optimisation of the robot as an automated inventory solution focuses primarily on the delivered accuracy. In this sense, the contribution of the RFID-driven navigation control is analysed and proved essential for an inventory with optimised accuracy and duration.

The autonomous navigation of the robot is not explicitly treated given it is not a novel contribution. However, the successful integration, configuration and parametrisation of algorithms are implicitly proved by the fact that inventories were completed without major incidents, after some initial iterations and adjustments. The validation of both interaction and operation procedures at inventorying are addressed in Chapter 4.

The simulation of the robot logic was set up in Gazebo[1], a robot simulation tool. Gazebo offers a complete toolbox for the simulation of robots in realistic scenarios. However, the simulation of RFID systems is not realistic. The reason is that the propagation of radio waves strongly depends on the specific characteristics of the environment and includes the

[1] http://gazebosim.org/

materials' electrical properties. Moreover, the simulation of radio wave propagation requires a complete finite element model. All in all, the simultaneous simulation of a high number of radio waves - the robot should operate in environments with thousands of tags - is in practice not possible. For all this, the simulation of RFID propagation was done with a simplified probabilistic model. Consequently, the simulation was not realistic in what regards RFID identification. Nevertheless, it was insightful in terms of navigation logic. The packages for a basic simulation of the inventory robot have been made available to the community[2].

5.2.1 Verification setting

The verification of any device requires an appropriate test bed, one where testing can be undertaken in conditions under control and reproduced as needed. In the robot case, the setting was expected to comply with a number of specific conditions that made the verification realistic and challenging at the same time. Pursued conditions were the following:

- Real environment
- Significant density of items
- Items challenging for RFID identification
- Perpetual inventory available

In this sense, the most appropriate accessible test bed was found to be the Pompeu Fabra University Library. It is an actual real environment with items placed in a packed manner. Books are known to be challenging for RFID identification due to the high amount of water in paper's composition, which absorbs radio waves. Plus, the Library Management System (LMS) keeps a record of available items. Altogether, the University Library was the perfect candidate but for one detail. The books were not originally RFID labelled. Therefore, RFID labels had to be coded and placed in books.

[2]https://github.com/UbiCALab/advanrobot

A representative aisle was chosen and 3,054 books labelled, including not only those in the aisle but also in neighbouring aisles, which results in a higher item density. The labels were placed on the last pages of the books. Compared to placing them on the spine, this is more challenging given the robot antennas are perpendicular to books pages when traversing an aisle. Generally, identification is easier when a tag and querying antenna are on parallel planes.

Figure 5.5 shows the robot in action at the verification setting and the characteristics are summarised in Table 5.3.

Figure 5.5: The robot at the verification setting, the University Library. Side covers removed so six RFID antennas are visible on one side.

Merchandise		Books
Number of items		3,115
Typical aisle width	$[m]$	1.05
A	$[m^2]$	12.0
σ_{items}	$[items/m^2]$	260
L_{aisles}	$[m]$	5.0
Intricacy	$[m/m^2]$	0.42

Table 5.2: Characteristics of the verification setting.

Books coding

The assignment of a unique item identifier to each of the books was addressed before the coding. An open coding Tag Data Standard[3] defines the structure of the Electronic Product Code (EPC), which supports diverse coding schemes. Among them, the most appropriate for the case in scope is the Serialised Global Trade Item Number (SGTIN), which is constructed from the combination of a product identifier (GTIN) and a unique serial number.

A GTIN is usually prefixed by a country code. In the case of books the GTIN is formed by prefixing the International Standard Bookd Number (ISBN) code with 978, the 'books country' identifier. The last digit is the Cyclic Redundancy Check (CRC).

The unique serial number is taken from a book's unique identifier that the library uses internally within their LMS. It is a 10 digit code that identifies uniquely a book and is printed on a label on its cover.

Following a standardised coding scheme implies that any RFID system can retrieve the information embedded in a product's EPC. Specifically, the coding scheme applied was an SGTIN-96, which uses 96 bits. Figure 5.6 depicts the coding scheme as defined in the standard. Note that the GTIN is the combination of the fields *GS1 Company Prefix* and *Indicator/Item Reference*.

The coding procedure was undertaken using an RFID encoding station[4] that consists of a barcode reader and an RFID tag encoder. The station was configured to generate the SGTIN code after reading a books barcode unique identifier and consulting the library database to retrieve the corresponding ISBN. At the same time, it checked whether a book belonged to the current library shelf. In this manner, misplaced books were uncovered.

Out of the initial 3,054 books coded, 122 were reported anomalous by the coding system. This accounts for a 4% of wrongly placed or erroneously informed references in the system. Of these, 2% were found

[3]https://www.gs1.org/standards/epc-rfid

[4]https://www.keonn.com/systems/tag-encoding/rfid-tag-encoding-systems.html

Scheme	SGTIN-96					
Logical Segment	EPC Header	Filter	Partition	GS1 Company Prefix (*)	Indicator (**) / Item Reference	Serial
Logical Segment Bit Count	8	3	3	20-40	24-4	38
Coding Segment	EPC Header	Filter	GTIN			Serial
URI portion		*F*	*C.I*			*S*
Coding Segment Bit Count	8	3	47			38
Bit Position	$b_{95}b_{94}...b_{88}$	$b_{87}b_{86}b_{85}$	$b_{84}b_{83}...b_{38}$			$b_{37}b_{36}...b_0$
Coding Method	00110000	Integer	Partition *Table 14-2*			Integer

Figure 5.6: SGTIN-96 coding scheme, the one used to code books in the library setting. Source: https://www.gs1.org/

to belong to different library facilities, misplaced not within the library building. In reality, such books are considered lost and unavailable, since they are actually not findable. This is of high significance to understand the value of an automated inventory and location solution, which prevents such situations.

5.2.2 Verification

The verification focuses on analysing the delivered accuracy under varying RFID-driven navigation control settings. The verification goal is twofold. First, proving that the robot can actually output inventories with an accuracy higher than 99%. And second, demonstrating that a navigation control based on the progress of identifications significantly improves both accuracy and efficiency.

For that, the robot is set up for inventorying in the verification test bed under three representative configurations for comparison. The comparison aims at assessing the contribution of RFID-driven interruptions and the fact that the robot twists in place when it stops due to new identifications.

Configuration	$th_{Journey}$	th_{Twist}	$Twisting$
A	-	-	-
B	0	1	False
C	0	1	True

Table 5.3: Configurations of the RFID-driven navigation control verification tests.

In configuration A, there are no RFID-driven navigation interruptions, which means that the robot traverses the verification aisle at cruise speed without stopping. Configuration B includes interruptions by setting thresholds to the most conservative combination possible. This implies stopping when just 1 new identification per second is registered and resuming the navigation only when identifications fall down to none. In configuration B, the interruptions do not trigger twisting in place but just stopping. And configuration C extends the former by including the twisting after an interruption.

The robot was configured to traverse the target aisle back and forth two times under each configuration. Altogether, each test consisted of four aisle passes. The inventory at each pass is the addition of items identified during the current and former passes. In this way, the contribution of traversing an aisle repeated times was assessed as well. Table 5.3 summarises the configurations.

On site, the verified baseline was computed with the support of an RFID handheld reader and the list of alleged books retrieved from the LMS. Handheld inventories consisted of four passes. Seven handheld inventories were used in the verified baseline computation. The same as with the robot.

Figure 5.7 and Table 5.4 show the verified accuracy values at each pass for the different configurations. The accuracy at each pass is the accumulation of former passes. Seven tests were completed under each configuration and the mean and deviation of results is assessed. The primary conclusion is that an accuracy higher than 99.0% can be achieved only if the navigation is adjusted following RFID identifications. As well, the contribution of twisting to the accuracy is revealed essential to reach

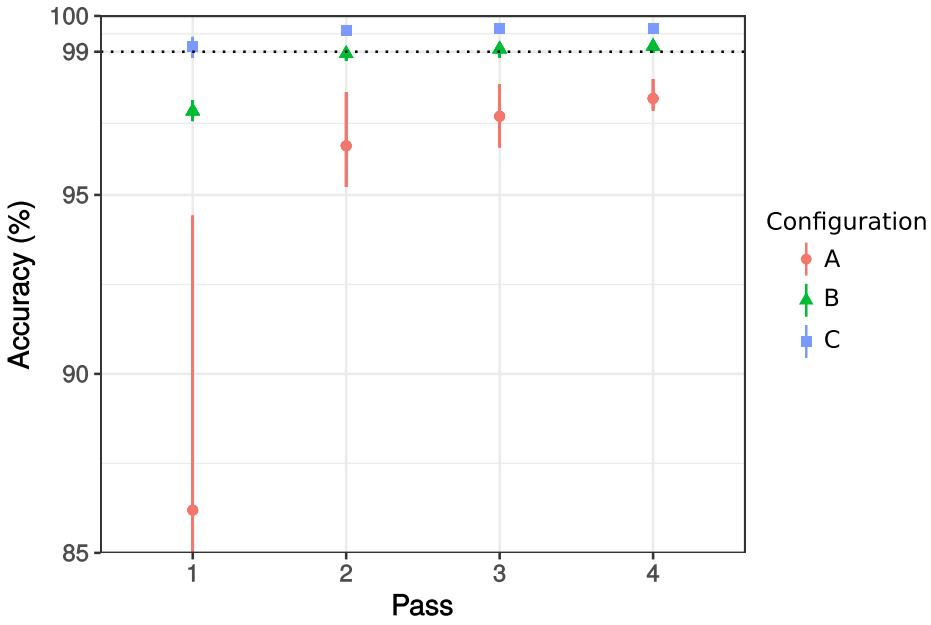


Figure 5.7: RFID-Driven navigation control verification tests results. Three different configurations were tested, which are summarised in Table 5.3. Each configuration involves four aisle passes. Each pass contributes to the accuracy in a cumulative way, new identifications are added to former passes. Seven tests were completed under each configuration and the mean (symbol) and span (lines) of the accuracy are assessed. The dotted horizontal line at 99% marks the minimum acceptable accuracy.

the required accuracy consistently. A further observation is that, whether in a single pass the accuracy is slightly higher than 99.0%, after a second pass it can reach up to 99.5%.

Among the assessed configurations, configuration C and two passes is selected as the optimal since it yields the maximum accuracy measurements. The contribution of a third and fourth pass are negligible.

A subsequent analysis is comparing the performance of the robot under this configuration to the handheld. Table 5.5 shows accuracy and duration results for comparison. Unsurprisingly, the robot and the handheld show similar figures in accuracy. It is important noting that handheld inventories were taken very thoroughly aiming at a relevant contribution

Configuration	Pass	mean(Acc)[%]	max(Acc)[%]	min(Acc)[%]
A	1	86.2	94.4	74.9
	2	96.4	97.9	95.2
	3	97.2	98.1	96.3
	4	97.7	98.2	97.4
B	1	97.3	97.7	97.1
	2	98.9	99.1	98.8
	3	99.1	99.3	98.8
	4	99.1	99.3	99.0
C	1	99.2	99.4	98.8
	2	99.6	99.7	99.5
	3	99.6	99.8	99.6
	4	99.7	99.8	99.6

Table 5.4: RFID-Driven navigation control verification tests results. Three different configurations are shown, which are summarised in Table 5.3. Each pass contributes to the accuracy in a cumulative way, new identifications are added to former passes. Accuracy Acc refers to the verified accuracy.

to the baseline. In this sense, handheld inventories can be considered of a very high quality. On the contrary, equivalent inventory durations were not expected before the optimisation of the navigation control. Nonetheless, it can be anticipated that robot and handheld can undertake comparable inventories in what regards accuracy and duration.

		Robot	**Handheld**
mean(Acc)	[%]	99.6	99.3
max(Acc)	[%]	99.7	99.7
min(Acc)	[%]	99.5	99.0
mean(ΔT)	[s]	601	598

Table 5.5: Robot (Configuration C, 2 passes) and handheld (4 passes) figures of accuracy and duration measured during the verification tests. Accuracy Acc refers to the verified accuracy.

5.2.3 Optimisation

Following verification, a set of tests are conducted with the ultimate goal of finding the pair of navigation thresholds that optimise inventory duration. Resuming the navigation only when no new identifications at all are received, although intuitively appropriate, may be too restrictive in terms of inventory duration. Increasing pairs of values are assigned to the pair of thresholds that determine the behaviour of the RFID-driven navigation starting from the most restrictive, or thorough, which is $(th_{twist} = 1, th_{journey} = 0)$. Note that such pair implies stopping and twisting after 1 or more new identifications and resuming the journey only when no new identifications are registered. The pairs of values are iteratively doubled until the accuracy falls below an acceptable level. Under each configuration, the robot was configured to complete 4 consecutive aisle traversals. An optimisation test includes the items accumulated during the 4 passes. Each test was performed 7 times and their mean and span assessed. For comparison, the verified baseline and subsequently accuracy were computed. Figure 5.8 shows the results for the pairs of thresholds analysed.

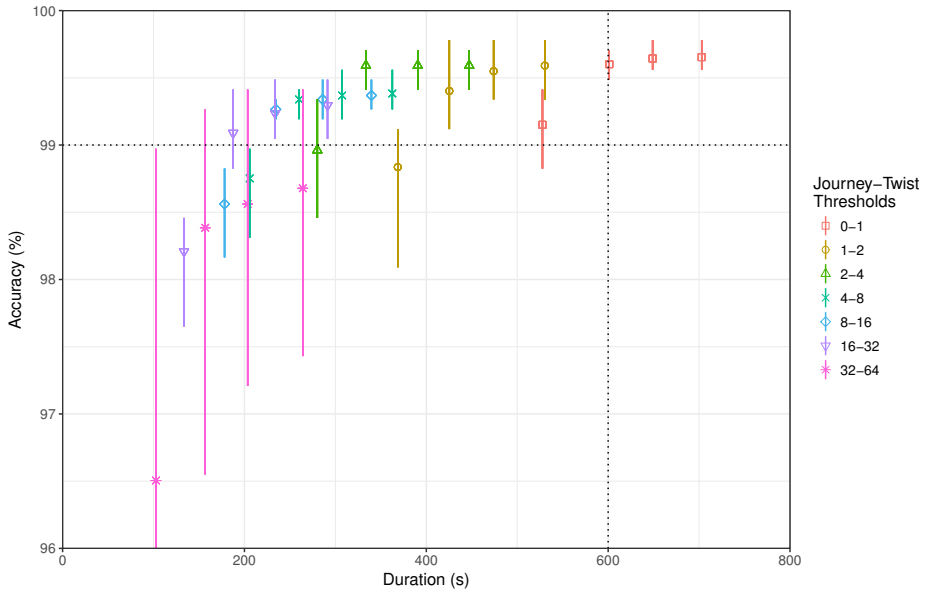


Figure 5.8: RFID-Driven navigation control optimisation results. The optimisation focus is in the comprise between inventory accuracy and duration. Each pair of thresholds involves four aisle passes, marked by four consecutive symbols. Each pass contributes to the accuracy in a cumulative way, new identifications are added to former passes. Seven tests were completed under each configuration and the mean (symbol) and span (lines) of the accuracy are assessed. The dotted horizontal line at 99% marks the minimum acceptable accuracy. The dotted vertical line at 600 s marks the baseline duration, the best measured during verification tests, $(th_{twist} = 1, th_{journey} = 0)$.

The foremost conclusion from optimisation tests is that inventory duration can be significantly reduced without compromising the accuracy by adjusting the navigation thresholds. The pair $(th_{twist} = 2, th_{journey} = 1)$ delivers the same accuracy as the pair $(th_{twist} = 1, th_{journey} = 0)$ in 30% less time. Both in a second pass. More interestingly, inventory duration can be more than halved if few fractions of accuracy can be dismissed. In two passes, the pair $(th_{twist} = 16, th_{journey} = 8)$ yields a 99.3% accuracy in just 235s which is 60% faster than the initial thresholds.

Considering the thresholds that optimise the performance of the robot, Table 5.6 lists side by side figures of merit for both the robot and handheld at its best pass for comparison. It is worth noting that, even the robot travels twice as much distance as the handheld, its effective speed is nearly three times faster. Also, the robot's reading capacity, expressed by the effective read speed, is 2.5 times higher. In this terms, one robot is equivalent to 2.5 effective people. In other words, the efficiency of people is 40% lower than the robot's. This is an expected figure since the robot includes three RFID readers and twelve antennas while a single handheld includes just one reader and one antenna. In conclusion, the robot as an automatic identification device does not only imply the automation of inventory taking but also enhancing the efficiency.

		Robot	**Handheld**
$Acc_{verified}$	$[\%]$	99.3	99.3
ΔT	$[s]$	235	598
L_{aisles}	$[m]$	11.0	5.0
v	$[m/s]$	0.047	0.008
v_{eff}	$[m/s]$	0.021	0.008
rs_{eff}	$[items/s]$	13.2	5.2

Table 5.6: Figures of merit of the optimal robot configuration and the handheld in the library. The optimal robot configuration is the one with $(th_{twist} = 16, th_{journey} = 8)$ and two passes.

To sum up, verification and optimisation tests disclosed a number of significant facts. Firstly, that the robot navigation should necessarily be

commanded by the progress of RFID identifications. Otherwise, the required accuracy figure is not achieved. Secondly, that a proper adjustment of the RFID-driven navigation thresholds is paramount for an efficient inventory taking. Importantly, optimal thresholds are expected to vary across environments with different characteristics. Hence, their initial adjustment given a specific target should be addressed. Lastly, that the robot is superior to a manually driven handheld as an automatic identification device.

5.3 Validation

The validation of the robot as an actual solution for the automation of inventories consisted of several experimentation periods at major retailers. In some other cases, the experimentation extended for longer periods, as part of a pilot. Either cases implied limitations stemmed from interfering in a real scenario. For instance, we were suggested to perform experiments at the hours of the day with less customers visits, or during closing hours. This presented notorious time constraints. As a consequence, in some cases, experiments could not be iterated or always completed consistently. Another usual limitation was not being provided with a perpetual inventory or product type references, mainly due to restricted access to retailers networks. Therefore, experimentation was adapted in each case to retailers particularities.

In most of the retailers the measured accuracy was higher than 99%. Nevertheless, in some cases it was not. In that cases, a thorough investigation of the items missed by the robot was conducted. The usual finding was a significant discrepancy between inventory figures reported by retailers and the actual physical stock. The justification included three variables. First and foremost, that the information retailers keep in their inventory systems is not accurate. Second, that it is not clear to retailers which sort of information they keep in their systems. For instance, which counts correspond to the shop floor and which to the back store. Third, that retailers are not knowledgeable of the internal procedures that affect

stock positions, for instance, products moved between locations and not subsequently reassigned to new locations. All in all, discrepancies arisen from the robot inventories served as reference for the investigation and clearance of inconsistencies. This was of significant importance since it disclosed a new application of the robot: the audit of other inventory systems.

Table 5.7 summarises the most significant experimental actions undertaken at actual retailers.

The thorough analysis of validation experimentations focuses on a specific retailer and period. The selected validation tests involve the unsupervised operation of the robot and the handhelds by actual store associates for an extended period of time. In this manner results are consequence of real store operations. Importantly, handheld inventories were already part of weekly associates tasks, which implies accumulated experience thus an expected quality. On the contrary, training was provided for the robot's operation. Hence, the robot's performance is benchmarked against the state-of-the-art stocktaking procedure.

Three zones within the store were selected following the retailer's experienced challenges and interest in inventorying them accurately and periodically. Table 5.8 shows the characteristics of the selected zones. The merchandise involved in the three zones are garments generally shaped in many combinations of models, colours and sizes that are not easy to discern from each other by the naked eye. For instance, women's underwear involve both a cup and a torso sizes and a great variety of models and colours are offered. Jeans involve as well two sizes, length and waist, and many different pants shapes not easy to distinguish at a glance. In these cases, few units of each product reference can be kept as garments become quickly densely packed and to occupy space. Because this, they are generally the products with a higher incidence of stock outs. Therefore, there is a critical need of inventory accuracy in those items.

Few initial considerations regarding validation are worth taking into account. The main pitfall is the absence of a perpetual inventory record due to confidentiality. However, product reference information was granted. Also, there were no inventory iterations on the same day, not even on con-

Retailer	Date	Weeks	Results
A	’15 Apr	1	First on site testing with the prototype. Accuracy >99% in two zones.
	’15 Aug	9	First on site testing with the early design. Accuracy >99% in three zones.
	’16 Aug	12	Complete store inventory: 7500^2 in 24 hours Robot operation unassisted by associates. Accuracy >99% in several zones.
B	’15 Mar	1	Accuracy >99% in one zone
	’15 Jun	5	Complete store inventory. Due to a significant inaccuracy in handheld and perpetual inventories, the robot accuracy was meaningless.
C	’15 May	3	Accuracy >99% in several zones.
	’18 Aug	1	Complete store inventory: 4000 m^2 in 6 hours Accuracy >99%.
D	’15 Dec	1	Accuracy >99% in one zone.
E	’16 Jan	1	Gathering a dataset for location.
	’18 TBD	8	Location assessment.
F	’16 Apr	3	Accuracy >99% in a mock-up store. Operational training.
	’18 Jul	8	Unassisted operation in production.
H	’17 Feb	4	Accuracy >99%. Robot operated by retailer’s personnel.
I	’17 Mar	1	Accuracy >99%. Robot operated by retailer’s personnel.
J	’17 Jun	1	Accuracy >99%. Robot operated by retailer’s personnel.
L	’18 Feb	8	Unassisted operation in production.
M	’18 Sep	8	Unassisted operation in production.

Table 5.7: Main robot field experimentation periods at retailers.

		Zone A	Zone B	Zone C
Merchandise		Jeans	Men's dresses	Women's underwear
Number of items		2,300	6,000	16,500
Typical aisle width	$[m]$	0.95	0.80	0.65
A	$[m^2]$	140	262	232
σ_{items}	$[items/m^2]$	16	21	71
L_{aisles}	$[m]$	72	160	149
$Intricacy$	$[m/m^2]$	0.51	0.57	0.64

Table 5.8: Characteristics of the store sections where experimental tests were conducted.

secutive days. As a result, the computed baseline was the filtered baseline (Eq. 5.3), consisting of a single robot and a single handheld inventory passes. One must note that the resulting filtered accuracy measure is a good device-to-device comparison indicator. However, it is not necessarily accurate in what regards physical stock. Furthermore, handheld data provided by the retailer did not include timestamps and the duration could not be computed individually for each handheld pass. Alternatively, an average duration of handheld scans was informed by the retailer.

During the validation period the robot completed a total of 32 inventory rounds. Out of those, 14 coincided with a handheld inventory on the same day and zone, becoming the ones valid for assessment. Although expected coincidences were higher, inventories were completely managed by store associates and deviated from schedule due to live changes in resources allocation and priorities. In any case, coinciding data sets are representative.

Figure 5.9 encloses a set of sub-figures that display all the relevant data for the comparative analysis of inventories. The first row of sub-figures displays the layout characteristics. The second and the third row display measured and computed inventory figures, including the filtered accuracy. Each sub-figure displays zone-wise measures for both the handheld and the robot. In the robot case, the measures are shown in average

and range of values. Otherwise, the handheld measures that involve the duration are just shown in average, given this was the data obtained from the retailer.

Looking at the layouts characteristics, it is worth noting that from zone A to zone C items density and intricacy show a growing trend. The growth in intricacy is reasonably due to aisles becoming narrower. In fact, in zone C the typical aisle width is below 0.70m, the robot's nominal navigation width. Although negative consequences can be expected, namely a slower navigation and a critical increase of the risk of getting stuck, this is preferred over missing aisles. In the end, the main goal is demonstrating the accuracy of the robot at inventorying cluttered spaces. Given items density and intricacy are the two main indicators of a layout's complexity regarding inventorying, from zone A to zone C the challenge to inventory is considered increasing.

Figure 5.9g serves as a starting point for the comparative analysis of the accuracy. The primary observation is that the robot delivers a filtered accuracy higher than 99.0% - marked in the figure with the dashed red line - in all the cases as opposed to the handheld, that fails at achieving the required accuracy. Furthermore, the robot's accuracy range throughout all the iterations is $[99.4\%, 100\%]$, which demonstrates a high robot precision. On the contrary, the handheld barely reaches the required accuracy and is very imprecise. On one hand, low handheld accuracies can be explained by the fact that the robot as an automatic identification tool is superior. On the other hand, the handheld imprecision is likely due to associates oversights, which are common at doing any repetitive and cumbersome task. In fact, the lowest bounds in handheld accuracy are likely due to unintentionally skipping a part of the zone. In the same sense, the robot is not error free and incurs failures. However, in the robot's case, oversights are very unlikely if not impossible. Robot errors are of different nature, yet most of them traceable and some recoverable. In this regard, Figure 5.9 does not include unsuccessful and self-reported inventory attempts by the robot. A last interesting trend is the decreasing handheld accuracy from zone A to zone C. The most likely reason is the increase in layout complexity.

Regarding duration, the time it takes the robot to complete a zone is comparable to that of a handheld (Fig. 5.9d), which complies with the initial requirement. Noteworthy, at the simplest zone (zone A), an associate with a handheld completes the inventory quicker than the robot. This reveals that associates are more efficient in uncomplicated zones. On the contrary, at inventorying zones of higher complexity, the robot is quicker. The former is better expressed by the effective speed (Fig.5.9h), which is normalized to the accuracy. In zone A the associates' effective speed is higher than the robot's, while the trend is inverted in subsequent zones. While the robot's travelled distance is higher in all the zones, the combination of duration and accuracy correction make its effective speed surpass that of the associate with a handheld. Interestingly, the effective speed trend is correlated to the Intricacy (Fig. 5.9d), which confirms the fact that the more complex a zone, the more efficient becomes the robot compared to doing the task manually.

Figure 5.9e shows that the robot travelled distance increases with intricacy while the handheld's distance remains proportional to the aisles length. This can be explained by a person's enhanced maneuverability. First, a person can traverse narrower aisles. Second, a person can overcome or put aside unexpected obstacles, for instance a fallen garment on the floor. In the same situations the robot needs to seek and follow an alternative path, which implies doing a walk around, and consequently increasing its journey. Thus, a person is more efficient regarding distance travelled. Besides, the robot is consistently faster since it walks more meters than the handheld in equivalent times (Figure 5.9f). A confirmed trend is that intricacy is correlated to the robot's speed (Figures 5.9c and 5.9f). More interestingly, the person's speed seems to be correlated to the items density. The latter would confirm the superior power of the robot as an RFID system compared to a single handheld. The robot is able to simultaneously identify more items. Hence, it deals with higher densities comparatively faster.

Figures of merit are shown in Figures 5.9h and 5.9i. Looking at the effective speed, a person is faster at completing the simplest section due to its better spatial efficiency. The trend is inverted at facing an increase of

aisles length. The decrease of effective speed for the robot is correlated to the intricacy while the handheld is affected by the increase in aisles length (zone B) and density (zone C). The effective read rate is computed as the amount of identified items per time unit and gives an idea of the identification capacity. The robot effective read rate is correlated to the density, which implies that the robot can assume the increasing density without compromising its pace. On the contrary, the number of identified items by the handheld decreases with density, expressed by the accuracy (Fig. 5.9g), and so does the effective read rate.

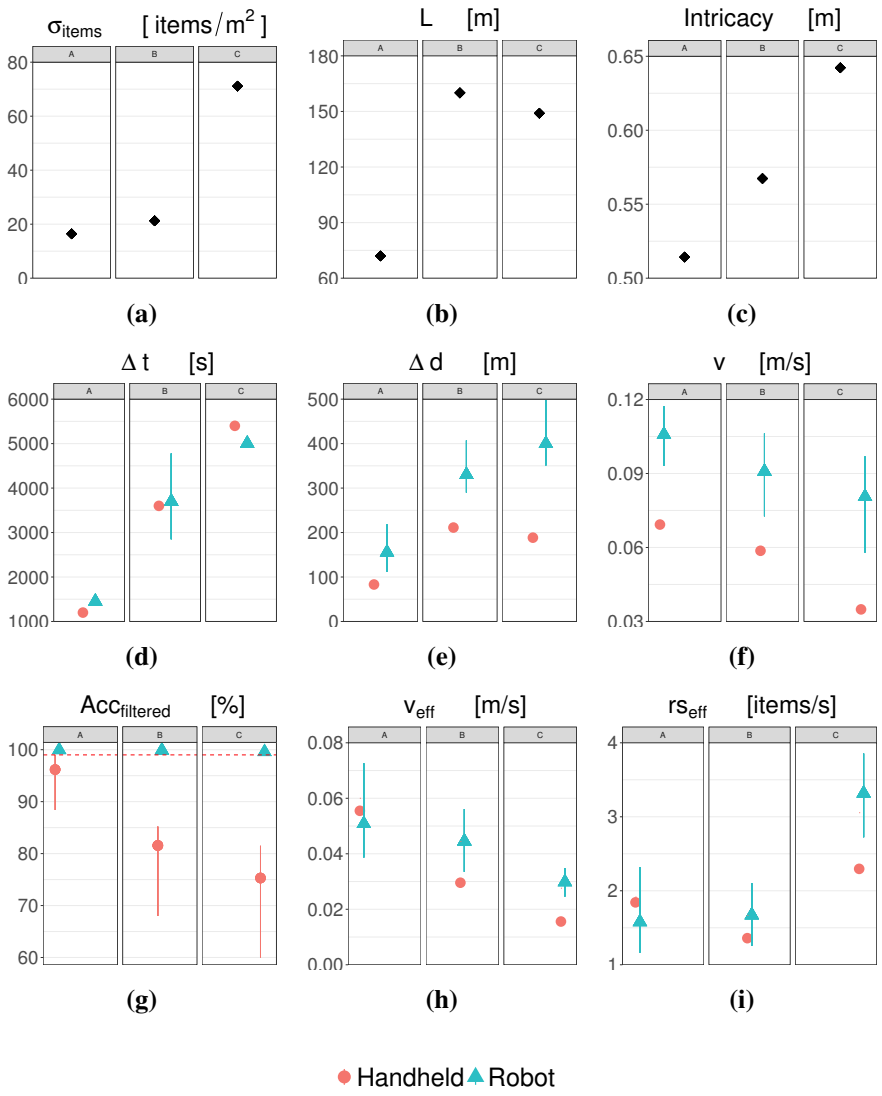


Figure 5.9: Summary of validation measures. On the first row, zone characteristics: (a) density of items, (b) aisles length and (c) intricacy. On the second row, inventory measures: (d) inventory duration, (e) inventory distance and (f) inventory speed. On the third row, figures of merit: (g) filtered accuracy, (h) effective speed and (i) read speed. Plots are actually a graphical display of tabular data.

Chapter 6

LOCATION OF ITEMS

The knowledge of the location of items is an essential part of inventory visibility, which aims at answering not only the *which* but also the *where* of products for an efficient supply chain and satisfactory service to customers. Basically, if an item's count appears positive in the inventory record but its location is not the expected and cannot be found, the item is in effect unavailable. The consequence is an unanticipated discrepancy between the inventory system count and reality. In other words, the lack of knowledge regarding items' locations contributes to the inventory record inaccuracy, which degrades the efficiency of operations. Moreover, the location of items unlocks complementary applications. For instance, the guidance of customers and associates to searched products, an efficient planning of product picking in the preparation of online orders sourced from the stores, or analysing the impact of a product placement in the sales. For all this, the computation of the location of RFID labelled products is addressed.

The computation of location is approached using probabilistic methods. Albeit theoretically possible, a computation based on the physical characterisation of the RFID detections is unlikely. Foremost, it would require the characterisation of every specific environment given the propagation of radio waves is strongly dependant on the characteristics of a setting. In this sense, every single object would have to be modelled,

which is not an assumable task. Additionally, the computation would involve a finite element model for solving Maxwell's equations, which implies a huge computational cost. Contrarily, probabilistic methods that do not rely on the characterisation of the environment are proposed.

Two different approaches are proposed and analysed for the computation of location. The first approach relies on permanent reference tags with known locations. The basic idea is assigning locations based on the similarity of detections to the reference tags ones. The second approach tackles the location of items using a measured detection model. It relies on temporary reference tags, which are used to learn a robot's detection model in a supervised fashion. After, locations are computed by applying Bayesian updating.

In order to assess the performance of location approaches a data set is essential. With that purpose, a data set was prepared, which is described in detail and shared openly with the community. To the best of my knowledge, it is the first open data set available for the assessment of RFID location algorithms.

6.1 RFID location dataset

The creation of a dataset for the assessment of RFID location algorithms is an important contribution of this thesis. The dataset includes thousands of identifications, distributed in space, of items with known locations and all the usual parameters informed by a commercial RFID system. The dataset can be used to apply any location algorithm intended for mobile RFID systems and assess its accuracy. To the best of our knowledge, to this thesis publication date no open dataset with a similar purpose was published. The data set is available at Zenodo[1] named ”RFID location dataset” [50].

The creation of the dataset involved two main steps. In the first place, the preparation of the setting, which extends the University Library test bed introduced in Chapter 5. The initial RFID labelled books were com-

[1] https://zenodo.org/

plemented with new ones. As a result, the setting encompasses 7,000 RFID labelled books distributed arbitrarily on shelves. An exception is the aisle initially labelled, which presents a higher density of books. Figure 6.1 shows the layout of the Pompeu Fabra University Library where the RFID location dataset was gathered.

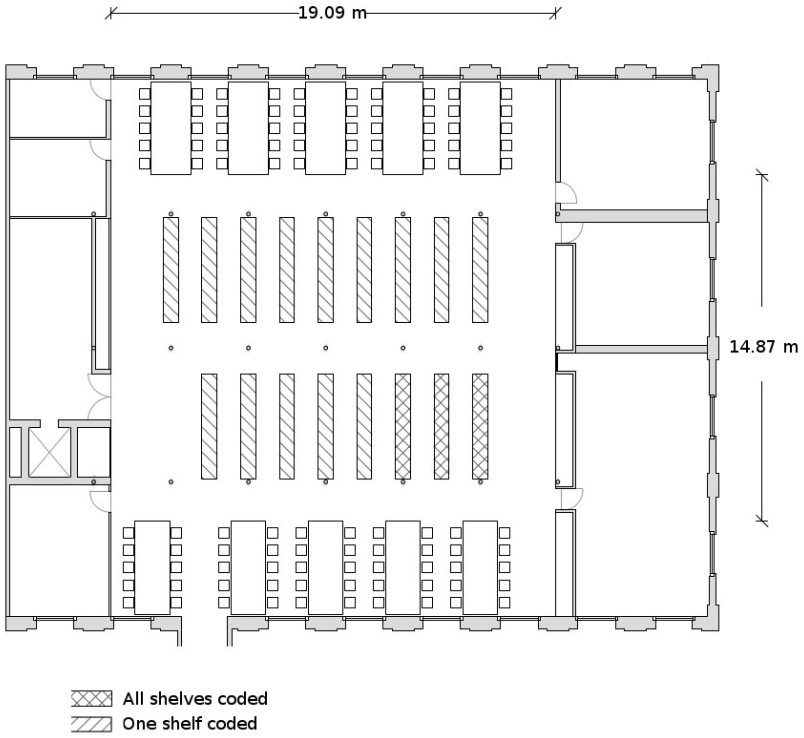


Figure 6.1: University Library layout where the RFID location dataset was gathered.

The library layout consists of aisles delimited by shelf racks. Each rack is organised in a number of rows and columns and their intersection is named a block. A block measures are 0.90x0.32x m. Figure 6.10 shows a shelf rack divided in blocks. The ground truth of the books locations was computed, rather than directly measured, by extending the locations of the first book in each block. These were indeed measured and noted down by hand along with the location of the last book in the block. Benefiting

from the fact that books are allegedly sorted by their descriptor, books are assigned a location by cumulatively adding the average book spine width to the known location of the first book. The block is finished when the last book in the block is found. The computed location ground truth is valid as long as the Library does not modify the organisation of books in shelves.

Subsequently, the robot was configured to complete four autonomous inventory rounds. A manual-guided inventory was also recorded, with the aim of gathering detections at unusual poses during autonomous inventory. Note that each book is identified from a number of robot poses during an inventory round. This means that for each book there are a number observations distributed in space.

It is worth noting that the configuration of the RFID system was not the same in all the inventory rounds recorded. Three inventory rounds were run under RFID session S1 and the other two under session S2. Justification comes from the fact that, in general, under session S2 less detections, thus less spatial diversity, are registered given RFID tags are quieted for longer time after their detection. Consequently, algorithms that exploit the spatial diversity of detections for location are likely to be penalised. On the other hand, more detections are registered under session S1 and, at the same time, more interferences occur, which can compromise inventory accuracy. In conclusion, while session S2 contributes to a higher accuracy in inventory, session S1 can contribute to higher accuracy in location. Providing data sets under both sessions grants the possibility of conducting a complete analysis. In this chapter the focus is on location and so the configuration used is session S1. In Chapter 5 the focus is on inventory and so the configuration used is S2.

The location dataset can be classified in two types of data. One, the data of static nature, which includes all the data that does not vary across inventory rounds. The second, dynamic data, the actual identifications and robot poses, which are specific to an inventory round.

Table 6.1: The location dataset consists of five inventory rounds under different configurations. Configurations using session S1 yield a higher number of identifications per book than configurations using session S2. This implies a higher diversity of robot positions, which is assumed to contribute positively to location algorithms.

	Type	Session	Average identifications per book
inv_1	Auto	S1	29.1
inv_2	Auto	S2	4.8
inv_3	Auto	S2	5.0
inv_4	Auto	S1	34.8
inv_5	Manual	S1	25.8

Static data

Static data includes the context of the dataset, a common spatial reference for spatial data. With this purpose, a 2D map of the environment is used. The 2D map is the one used by the robot for navigation, introduced in Chapter 4. The 2D map consists of two files: a bitmap that contains cell occupation information; and a description file that details the map's origin and resolution. Both books and the robot poses during the inventory are referenced to the 2D map. Additionally, a 3D occupancy map is provided separately. The 3D map can be exploited to delimit the possible locations of books since an object is necessarily an occupied cell in the search space. The 2D and 3D maps share a common origin. Static data also includes the transforms between the robot's origin, which determines the robot poses, and the antenna's coordinates. Using transforms one can compute the location of the antenna's at informed robot poses.

Static data also includes the baseline of books locations, which are referenced to the aforementioned map and computed as described earlier.

Dynamic data

Dynamic data consists of robot poses and identifications, which are specific to inventory rounds. Robot poses and identifications originate from two different computation sources which are time synchronised by means of the Network Time Protocol (NTP). Thus, both sources of data are related by a synchronised epoch. This is of utmost important to assign identifications the corresponding location of the antenna at the moment it was received. In this manner, the spatial source of each identification can be determined to contribute to algorithms that exploit the diversity of identifications in space.

Robot poses are referenced to the 2D map's origin and include the position and orientation of the robot during an inventory round. Identifications include all the usual parameters reported by an RFID reader and the descriptor of antenna that originated the identification. A commercial RFID reader usually reports RSSI, RF phase, RF frequency, a timestamp and an RF port identifier. The latter univocally determines the antenna that originated the identification.

All the robot poses during the navigation are provided, even not showing any associated identifications. Robot poses without identifications can be exploited by algorithms that make use of negative identifications to complement positive identifications.

Table 6.2 lists the features provided corresponding to each subset type.

6.2 Clustering identification streams

The first approach explored for the location of RFID labelled objects relies on the use of reference tags. The basic idea is placing objects in known locations and assigning those locations to other objects identified with similar characteristics. Intuitively, location with reference tags is a clustering problem.

The identification of an object (EPC) is defined by the different identification instances during an inventory round, which are distributed in time

Table 6.2: List of data included in the location dataset

Dynamic			
Robot poses	Timestamp Position Orientation	epoch cartesian quaternion	1510944342382526 (-2.73, 4.81, 0) (0, 0, -0.31, 1)
Identifications	Timestamp Antenna ID RSSI RF Phase RF Frequency	epoch id dBm degrees Hz	1510944342382526 reader-02.2 -78 149 867341

Static			
Book locations	Book ID Position	id cartesian	1005144591 (6.40, -0.60, 0.64)
Fixed transforms	Parent frame Child frame Translation Rotation	id id cartesian quaternion	base-link antenna-link (0.15, 0.17, 0.03) (0.7, 0, 0, 0.7)
2D map	Occupancy grid	bitmap	Figure 4.8
3D map	Occupancy grid	octomap	Figure B.1 (Annex)

and space given the extended reading reach of the robot. Figure shows the set of identification instances for a given object. Hence, the detection of an object, represented by its Electronic Product Code (EPC) or unique identifier, can be expressed by a stream of identification instances

$$ID_{epc} = \{id_0, id_1, ..., id_n\} \tag{6.1}$$

An identification instance id_i represents the detection of the object at an ith time instant, thus pose in space, and is determined by a set of features

$$id_i = \{epc_i, ts_i, ant_i, pose_i, RSSI_i, phase_i, freq_i\}$$

$$\begin{aligned} ant_i &= \{n \mid n \in \{1, ..., N\}\} \\ pose_i &= \{position_i, orientation_i\} \\ position_i &= \{x_i, y_i, z_i\} \\ orientation_i &= \{x_i, y_i, z_i, w_i\} \\ RSSI_i &\in [-85, ..., -30] \\ phase_i &\in [0, \pi] \end{aligned} \tag{6.2}$$

being epc_i the electronic product code of the item; ts_i the timestamp; ant_i the antenna that originated the identification; $pose_i$ the pose of the antenna at the moment of the identification; $rssi_i$ the received strength of the signal; $phase_i$ the phase of the received signal; and $freq_i$ the frequency of the signal carrier. These parameters represent two types of features: those that are related to the spatial characteristics of the identification; and those that represent the physical characteristics of the electromagnetic propagation of the signal or the RFID system identification parameters.

In order to assess the similarity of two streams of instances, an appropriate similarity or distance measure needs to be defined. One could think of using all the features of each instance, which implies a multidimensional and multivariate measure. Instead, the proposed measure aims at a simplification. The focus of the similarity is on the spatial variables and assumes that RF propagation features are secondary. The hypothesis is that a similarity measure based only on the spatial characteristics suffices to group the streams of instances into clusters. A further simplification relies on the fact that at a specific time all the instances have the same spatial characteristics. Hence, instead of computing a similarity based on the pose, just the timestamp is used for assessment, which transforms the multivariate and multidimensional initial problem to a univariate one. However, the timestamp alone represents a robot pose but not the specific antenna pose. For that, the simplified identification instance includes the

identifying antenna ant_i along with the timestamp ts_i and epc_i.

$$\widehat{id_i} = \{epc_i, ts_i, ant_i\} \tag{6.3}$$

Now, the representation of a stream of identifications for a given object can be reformulated.

$$\widehat{ID}_{epc} = \{\widehat{id_0}, \widehat{id_1}, ..., \widehat{id_n}\} \tag{6.4}$$

For convenience, the original identification stream can be reformulated by grouping identifications from a common antenna in sub-streams. Given each antenna sub-stream corresponds to a unique pair of EPC and antenna, their enclosed identifications carry epc_i and ant_i implicitly. Then, an antenna sub-stream can be expressed as a sequence of timestamps.

$$\widehat{ID}_{epc}^{ant_i} = \{\widehat{id_0}, ..., \widehat{id_P}\} = \{ts_0, ..., ts_P\} \tag{6.5}$$

A complementary expression of the former is obtained by binning timestamps. An empirical fact is that objects close to each other are not necessarily identified at the very same second but within a common time window. Hence, the similarity between streams can be better assessed using time bins of a given resolution. Each time bin encloses zero to many detections. For convenience, the final formulation is expressed by the cardinality of time bins or, which is the same, the amount of detections registered within.

$$\widehat{\widehat{ID}}_{epc}^{ant_i} = \{|ts_{0:T}|, ..., |ts_{(P-1)T:PT}|\}$$

$$|ts_{(k-1)T:kT}| \equiv \text{Cardinality of the } k_{th} \text{ time bin}$$
$$T \equiv \text{Time bin width} \tag{6.6}$$

Now, the identification stream can be expressed as the concatenation of antenna sub-streams, following always the same antenna order. To

do so, the binned timestamp information is transformed into an ordered sequence of bins. The temporal dimension is discarded and the identification stream becomes an ordered sequence of cardinals. Antenna information is not discarded, it is implicit given the sequence order is the same among different identification streams, thus the underlying time dimension and antenna are represented by the position of the bin in the sequence.

$$\widehat{\widehat{ID}}_{epc} = \widehat{\widehat{ID}}_{epc}^{ant_1} \| \widehat{\widehat{ID}}_{epc}^{ant_2} \| ... \| \widehat{\widehat{ID}}_{epc}^{ant_N} \tag{6.7}$$

$$\widehat{\widehat{ID}}_{epc} = \{|bin_0|, ..., |bin_P|, ..., |bin_V|, ..., |bin_{P+V}|\} \tag{6.8}$$

The formulation expressed by equation 6.8 is the actual one used for the computation of the similarity between streams. In fact, the aggregation of all the antennas represented by such formulation. One must note that bins from different antennas are independent even the underlying time window coincides. This is due to the fact that different antennas involve different poses relative to the identified object. Consequently, two identifications that happen at the same time but from different antennas do not contribute positively to similarity. In conclusion, an identification stream is expressed as the number of detections partitioned in time, for each robot antenna separately.

The graphical representation of identification streams are shown in Appendix A for two selected blocks of books and two different antennas. Each plot shows the identifications of a single antenna. Each line represents the identification stream of a single book. The similarity in the time domain between streams within the same block, and from the same antenna, can be recognized. It is apparent a jitter or offset between some of the streams within the same block, which is the intuition for binning instances. It is also apparent the feasibility of clustering if a proper similarity coefficient is defined.

6.2.1 Similarity coefficients

A set of similarity coefficients are considered for the quantification of similarity between identification streams. On one hand, the Battacharya coefficient, a known measure which approximates the overlap between statistical samples. On the other hand, a set of coefficients are defined for the specific purpose of clustering RFID identifications. Afterwards, the coefficients are transformed to distances and used to populate the distance matrix required for clustering. Similarity coefficients can be deemed a sort of identification streams correlation.

For convenience, onwards, the pair of simplified identification streams $\widehat{\widehat{ID}}_{epc1}$ and $\widehat{\widehat{ID}}_{epc2}$ is expressed as follows

$$\begin{aligned} \widehat{\widehat{ID}}_{epc1} &= R = \{r_0, ..., r_Q\} \\ \widehat{\widehat{ID}}_{epc2} &= S = \{s_0, ..., s_Q\} \end{aligned} \tag{6.9}$$

Battacharyya coefficient

The first coefficient analysed is the Bhattacharyya coefficient (BC) [51]. It is a measure of overlap between two statistical samples, widely used in statistics to assess the separability of classes. Basically, it partitions a series domain and integrates overlapping occurrences.

$$BC(R, S) = \sum_{i=0}^{Q} \sqrt{r_i s_i} \tag{6.10}$$

Note that in case there are no overlapping samples, the battacharya coefficient is zero, according to a null similarity. Note as well that the battacharya coefficient is dependent on the resolution of the partition. This is convenient in our application since identifications of near objects do not exactly happen at the same instants. However, the Battacharya coefficient is not a normalised measure.

Sample match coefficient

The sample match coefficient (SMC) measures the percentage of coincidence between identification streams bin-wise.

$$SMC(R,S) = \frac{1}{Q+1} \cdot \sum_{i=0}^{Q} \frac{\min(r_i, s_i)}{\max(r_i, s_i)} \tag{6.11}$$

SMC is null when no overlap occurs, it is 1 in case of a perfect match, and it is dependent on the bin width. As opposed to the Battacharyya coefficient this is a normalized measure.

Bin match coefficient

The bin match coefficient (BMC) considers bins either occupied or empty. The cardinality of each bin is ignored. It measures the percentage of coincidence in a binary way.

$$BMC(R,S) = \frac{1}{Q+1} \cdot \sum_{i=0}^{Q} [r_i \geq 1]\,[s_i \geq 1] \tag{6.12}$$

BMC is null when no overlap occurs and it is dependent on the bin width. A perfect match can be measured even the identification streams are not strictly equal. This measures aims at stressing out identifications within a time window as key events as opposed to quantifiable events. In case the cardinality of all bins is one, SMC is equal to BMC. As opposed to the Battacharyya coefficient this is a normalized measure.

Weighted intra-bin coincidences SMC

The weighted sample match coefficient (wSMC) extends SMC by applying a weighting factor w_i to each bin. The factor aims at giving a stronger relevance to those bins that expose a higher number of coincidences even not showing a perfect match. For instance, a bin coincidence with the likes of $r_0 = 1$ and $s_0 = 1$ is penalised (SMC=1, wSMC=0.5) while a

coincidence with the likes of $r_0 = 7$ and $s_0 = 10$ is favoured (SMC=0.7, wSMC=0.6).

$$w_i = \frac{\min(r_i, s_i)}{1 + \min(r_i, s_i)} \tag{6.13}$$

$$wSMC(R,S) = \frac{1}{Q+1} \cdot \sum_{i=0}^{Q} w_i \cdot \frac{\min(r_i, s_i)}{\max(r_i, s_i)} \tag{6.14}$$

wSMC is null when no overlap occurs and it is dependent on the bin width. A perfect match is only measured when identification streams are strictly equal and include an infinite number of occurrences. It is a normalized measure.

Weighted length of coincidences coefficients

The former coefficients SMC, BMC and wSMC can be adjusted by applying a factor that considers the length of coincidences. It aims at giving more relevance to pairs that involve a higher number of coinciding bins. For instance, a pair with the likes of $\{r_0 = 1, r_1 = 1, r_2 = 1\}$ and $\{s_0 = 1, s_1 = 1, s_2 = 1\}$ is favoured over a pair with the likes of $\{r_0 = 1\}$ and $\{s_0 = 1\}$. This is due to the fact that the first pair involves three coinciding bins while the second pair involves only one.

$$W(R,S) = \frac{\sum_{i=0}^{Q} [r_i \geq 1]\,[s_i \geq 1]}{1 + \sum_{i=0}^{Q} [r_i \geq 1]\,[s_i \geq 1]} \tag{6.15}$$

$$WSMC(R,S) = W(R,S) \cdot SMC(R,S) \tag{6.16}$$

$$WBMC(R,S) = W(R,S) \cdot BMC(R,S) \tag{6.17}$$

$$WwSMC(R,S) = W(R,S) \cdot wSMC(R,S) \tag{6.18}$$

Coefficients discussion

An illustrative example supports the discussion regarding coefficients. Figure 6.2 depicts four representative identification streams, each bin showing the amount of identifications within it.

Table 6.3 summarises the similarity coefficients of the example. For convenience, the self-similarity of an item with itself is computed as well. Looking at self-similarities, the BC reveals a strong dependence on the intra-bin coincidences, while the amount of coinciding bins is apparently residual. Note that the higher BC self-similarity is for $epc3$, which coincides only in two bins but in a high number (8) of intra-bin samples. Contrarily, BMC puts the focus in the number of coinciding bins. Note that the self-similarity is 1 in all the cases and the similarity between items is higher the higher the number of coincidences. SMC extends BMC by including intra-bin coincidences in the computation. While self-similarities are 1, equal to BMC, the similarity between $epc1$ and $epc2$ is no longer one given their bins carry a different number of intra-bin samples. The SMC between $epc1$ and $epc4$ is the same as its BMC given they carry the same amount of intra-bin samples. Next, the wSMC takes in consideration not only the intra-bin coincidence, but the amount of intra-bin coincidences. Focusing on self-similarities, while $epc1$ moves from an $SMC = 1$ to a $wSMC = 0.50$ given there is only one coinciding sample in each bin, $epc3$, which encompasses a higher number samples, moves from an $SMC = 1$ to a $wSMC = 0.68$. However, the self-similarity of $epc1$ and $epc4$ is the same $wSMC = 0.5$ even $epc4$ includes a higher number of coinciding bins. For this reason, a weight to the length of coincidences is included in $WwSMC$. The self-similarities are now a $WwSMC = 0.40$ for $epc1$ and a $WwSMC = 0.45$ for $epc4$, due to the higher number of coinciding bins in $epc4$. The other measures that apply a weight to the length of coincidences, $WBMC$ and $WSMC$ show equivalent trends.

R	S	BC	BMC	SMC	wSMC	WBMC	WSMC	WwSMC
epc1	epc1	4.00	1.00	1.00	0.50	0.80	0.80	0.40
epc1	epc2	6.90	1.00	0.58	0.29	0.80	0.47	0.23
epc1	epc3	5.66	0.50	0.06	0.03	0.33	0.04	0.02
epc1	epc4	4.00	0.40	0.40	0.20	0.32	0.32	0.16
epc2	epc2	14.00	1.00	1.00	0.68	0.80	0.80	0.54
epc2	epc3	13.86	0.50	0.38	0.32	0.33	0.25	0.21
epc2	epc4	6.90	0.40	0.23	0.12	0.32	0.19	0.09
epc3	epc3	16.00	1.00	1.00	0.89	0.67	0.67	0.59
epc3	epc4	5.66	0.20	0.02	0.01	0.13	0.02	0.01
epc4	epc4	10.00	1.00	1.00	0.50	0.91	0.91	0.45

Table 6.3: Similarity coefficients corresponding to the four items whose identification streams are depicted in Figure 6.2.

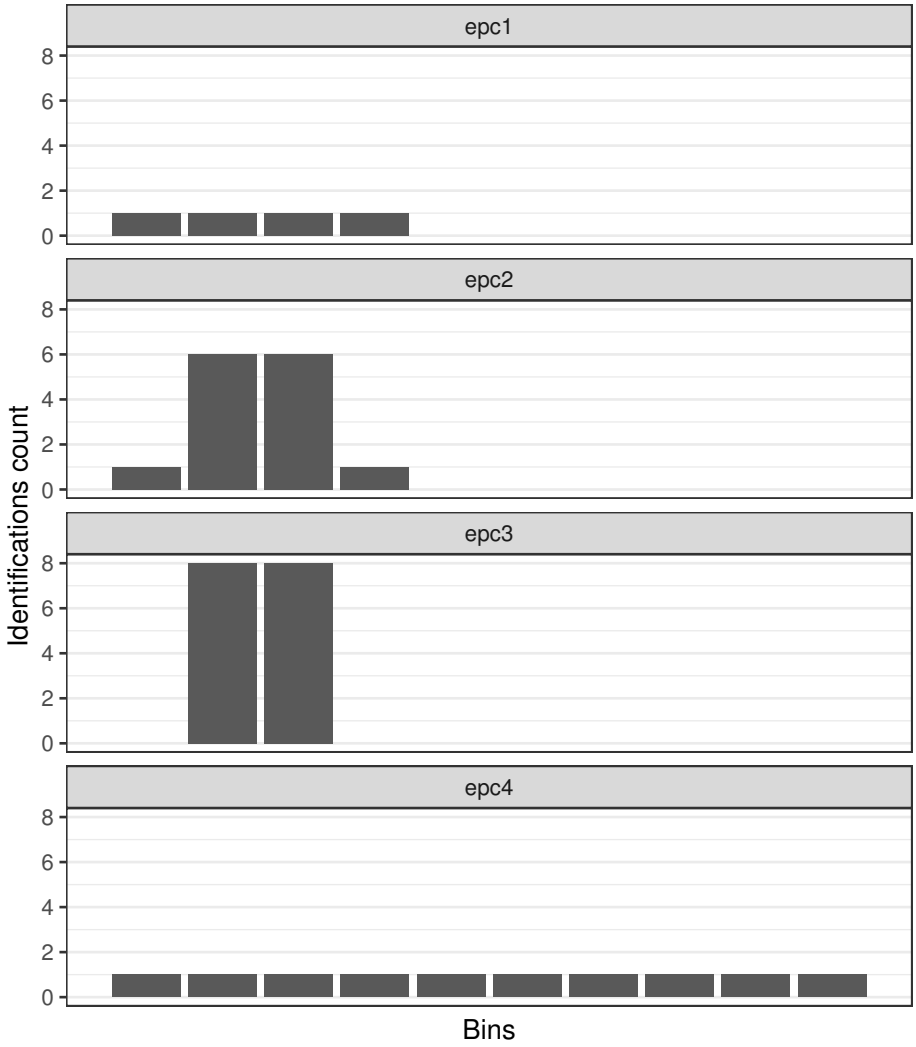


Figure 6.2: Representative binned identification streams that support the similarity coefficients discussion. The identifications’ count in each bin is depicted for four items.

Distances

Clustering requires a measure of distance between samples, the identification streams. The distance between two samples is complementary to the

similarity coefficient. On one hand, the Battacharya coefficient is transformed to a distance applying a normalization. The normalization factor is the maximum BC measured across all pairs of identification streams under assessment.

$$d_{BC}(R,S) = 1 - \frac{BC(R,S)}{\max(BC(\widehat{\widehat{ID}}_{epc_i}, \widehat{\widehat{ID}}_{epc_j}))} \tag{6.19}$$

On the other hand, similarity coefficients that are already normalized are transformed to distances following equation 6.20, which is shown for a generic match coefficient, expressed as XMC, and applies to all the other coefficients proposed.

$$d_{XMC}(R,S) = 1 - XMC(R,S)$$

$$XMC \in \{BMC, SMC, wSMC, WBMC, WSMC, WwSMC\} \tag{6.20}$$

6.2.2 Clustering

The clustering algorithm chosen is hierarchical clustering [52]. The main reason is the flexibility that hierarchical clustering offers in space partitioning. The output of hierarchical clustering is not just a set of clusters but the hierarchical proximity of samples. Its graphical representation is called a dendrogram, a tree-like structure in which the distance between samples or groups of samples is expressed by the length of the branches that connect them. By cutting the branches at a chosen level, clusters are formed. Figure 6.3 shows an example of a dendrogram that is cut at a given height, marked with a red dashed line.

In the library, one could choose to create clusters of blocks, shelves or book topics indistinctly by cutting the dendrogram at higher or lower heights. In a retail store or warehouse, it could be used to generate views

[2]https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/USArrests.html

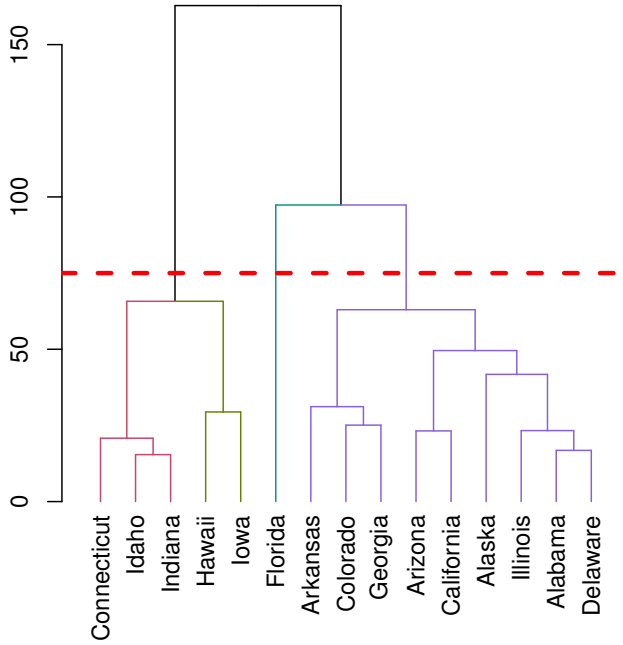


Figure 6.3: Dendogram of a subset of Violent Crime Rates by US State dataset[2]. The vertical axis shows the distance between samples grouped by the dendogram branches. The red dashed line marks the height at which the dendogram is cut to form 4 clusters, shown in different colours.

of products grouped at different granularities. In this sense, the choice of the clustering algorithm is mandated by its potential applications.

The clustering algorithm implementation used is *hclust*, included in [R][3] statistical computing software. The strategy is agglomerative: each sample starts as a one-member cluster and is recursively merged with near clusters, creating the hierarchy bottom-up. The distance between clusters

[3]https://www.r-project.org/

at each stage is measured based on the agglomeration method selected. The agglomeration methods available are detailed in Table 6.4.

Table 6.4: Agglomerative hierarchical clustering methods. A and B represent two clusters, $D(A, B)$ the distance between them, and $d(a, b)$ the distance between two members of a cluster.

Method	$D(A, B)$	
Single	$min\{d(a, b)\}$	$a \in A$, $b \in B$
Complete	$max\{d(a, b)\}$	$a \in A$, $b \in B$
Median (WPGMC)	$median\{d(a, b)\}$	$a \in A$, $b \in B$
Average (UPGMA)	$\frac{1}{\|A\|\|B\|} \sum \sum d(a, b)$	$a \in A$, $b \in B$
Centroid (UPGMC)	$\|c_A - c_B\|^2$	$c_A \equiv$ centroid of A
McQuitty (WPGMA)	Any[a]	
Ward D	$\frac{\|A\|\|B\|}{\|A\|+\|B\|} \|c_A - c_B\|^2$	$c_A \equiv$ centroid of A

[a] McQuitty refers to an update formula implementation that can use any distance measure [53].

6.2.3 Clustering evaluation

Clustering is assessed using a complete inventory round of the location dataset that includes 7,000, 256 book blocks and an average of 33 identifications per book. One must note that eight antennas are involved, four on each side of the robot. As introduced in Equations 6.7 and 6.8, a single identification stream merges the information from all the antennas. Antenna information remains implicit, hence it is taken into account at measuring similarities and computing the clustering.

The evaluation analyses combinations of three parameters: similarity coefficient, bin width and clustering method. Evaluation is approached in three steps. Initially, results of clustering only 5 blocks, or classes, are assessed. Each book block encompasses 31 books in average. The goal is figuring out the combination of parameters that suits best for clustering the books in the library. Subsequently, the combination of parameters that show a higher performance are applied to a subset that includes 50 blocks.

Table 6.5: Parameters and values that are combined for clustering evaluation.

Distances	BC BMC SMC wSMC WBMC WSMC WwSMC
Bin widths [s]	0.5,1,3,5,9 12,15,18,21
Clustering methods	Table 6.4

At this step, the goal is validating whether the clustering performance keeps up with an increased number of samples. To conclude, clustering is applied to the complete library dataset, which encloses 225 blocks and 7,000 books.

The values of the parameters that are combined for the assessment of clustering are listed in Table 6.5.

Evaluation measures

The measures used for assessment are general accuracy measures used in classification: Precision (P); Recall (R); and F1 score (F1). They are computed after the amount of samples correctly included in a cluster, or true positives (TP); the amount of samples incorrectly included in a cluster, or false positives (FP); the amount of samples correctly left out of a cluster, or true negatives (TN); and the amount of samples incorrectly left out of a cluster, or false negatives (FN).

$$P = \frac{TP}{TP + FP} \tag{6.21}$$

Table 6.6: Example of clustering two classes, A and B, and the corresponding confusion matrix.

Cluster A

a_1	a_2	b_1
a_3	a_4	

Cluster B

b_2	b_3	b_4
b_5	a_5	a_6

	Same cluster	Different clusters
Same class	TP=13	FN=12
Different classes	FP=12	TN=17

$$R = \frac{TP}{TP + FN} \tag{6.22}$$

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \tag{6.23}$$

In the particular case of clustering, TP, FP, TN, and FN are computed considering pairs of samples. An illustrative example with two clusters, Figure 6.6, is used for explanation. Cluster A encloses $\{a_1, a_2, a_3, a_4\}$, which implies $\binom{4}{2}$ pairs correctly formed. Cluster B encloses $\{b_2, b_3, b_4, b_5\}$, which implies $\binom{4}{2}$ correct pairs, and $\binom{2}{2}$ correct pairs due to $\{a_5, a_6\}$. The aggregation of correct pairs gives TP. On the other hand, the total amount of incorrectly formed pairs in Cluster A are the combinations of $\{a_1, a_2, a_3, a_4\}$ and b_1. Analogous in Cluster B, the combinations of $\{b_2, b_3, b_4, b_5\}$ and $\{a_5, a_6\}$ are the incorrect pairs. The aggregation of incorrect pairs gives the total FP of the clustering. Extending the calculations, the final confusion matrix is computed (Table 6.6), which provides the sources of accuracy measures P, R and F1. The example has $P = 0.52$, $R = 0.52$ and $F1 = 0.52$.

In addition, Purity, a clustering-specific measure, is used. The share of the predominating class in each cluster is computed and averaged among all clusters. From the example (Table 6.6), in Cluster A the predominating class is a_i and its purity is $\frac{4}{5}$. Cluster B has a purity of $\frac{4}{6}$, given the

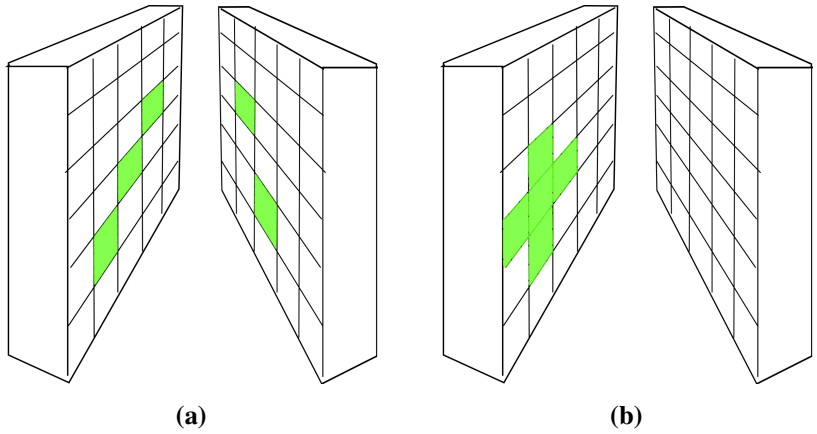


Figure 6.4: Schematics of an aisle perspective composed by two shelf racks divided in book blocks. Highlighted, (a) Subset A blocks and (b) Subset B blocks. Subset B is more challenging than Subset A for clustering because of the proximity of blocks.

predominating class is b_i and the total number of samples in the cluster is six. The overall Purity is the average of the former: $\frac{1}{2} \cdot (\frac{4}{5} + \frac{4}{6}) = 0.73$.

Subset A and Subset B (5 blocks)

In order to do an initial assessment of the different configurations, two subsets of books blocks (classes) are selected. Subset A includes five blocks located close to each other. This is considered an easy case for clustering. In this manner, the coarse separability of blocks is assessed. Subset B includes five adjacent blocks, this is in the same shelf rack and next to each other. In this manner, the separability of blocks under the most challenging configuration is assessed. Figure 6.4 shows schematics of the distribution on the library shelves of blocks in Subset A and Subset B.

Appendix A includes graphs of the identification streams of Subset A and Subset B for each book and two robot antennas. The graphical representation of identification streams supports the intuition and challenge of separability.

Results of clustering Subset A using the measures and methods pro-

posed are shown in Figure 6.5. A number of preliminary conclusions can be drawn. First and foremost, that clustering methods $centroid$, $complete$, $median$ and $single$ are futile. These methods are discarded in forthcoming analysis. One can observe that the best method is *ward.D2* for clustering Subset A. There is no similarity measure that outperforms the others. All the similarity measures at a bin width of 1 second provide a perfect clustering. This was unexpected since it contradicts the initial hypothesis that binning identifications could help. In practice, the time resolution of identifications is 1 second thus binning at 1 second is equivalent to not binning. The perfect clustering of Subset A is depicted in the form of a dendogram in Figure 6.6. The main conclusion from applying clustering to Subset A is that RFID identifications can be grouped by using a similarity that relies only in time coincidences. The initial hypothesis that identification streams can be compared looking at identifications timestamps is demonstrated.

Results of clustering Subset B are shown in Figure 6.7. The main conclusions drawn from Subset A apply as well to Subset B: clustering method *ward.D2* outperforms the other methods and the optimal bin width is 1 second. However, in this case the best result is not a perfect clustering. The maximum F1-score is 0.91 and the maximum Purity is 0.95. The dendogram resulting of clustering Subset B (Figure 6.8) shows that 4 out of 74 books are assigned to a wrong block. Although the result of clustering Subset B is not perfect, given that blocks are adjacent to each other, it is considered a good result. The separability of blocks using a similarity that relies on time coincidences is still valid.

Subset 50 (50 blocks)

After proving the feasibility of clustering five books blocks, the next goal is assessing the scalability of the method. For that, the number of blocks, thus books, is increased. The same experiments are conducted on 50 randomly chosen blocks, Subset 50, which accounts for 1250 books. Given the high number of books, no dendograms and identification streams are depicted in the subsequent analysis.

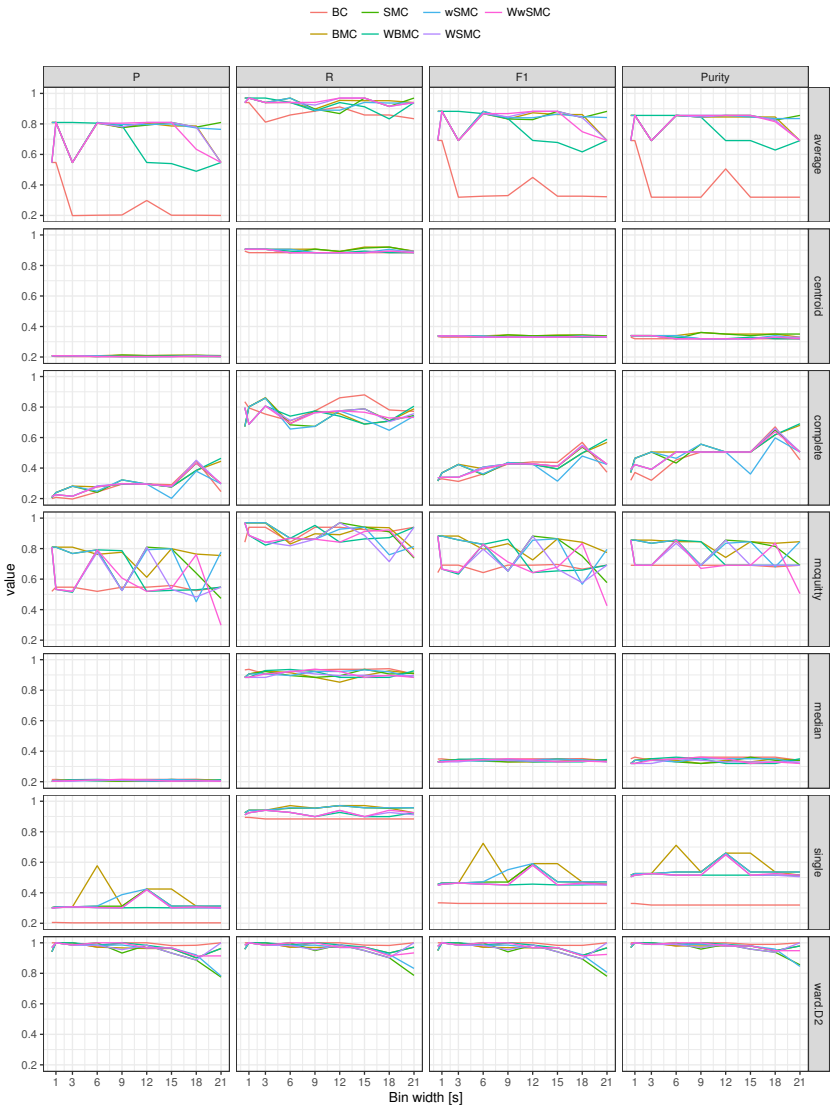


Figure 6.5: Results of clustering subset A book blocks (classes).

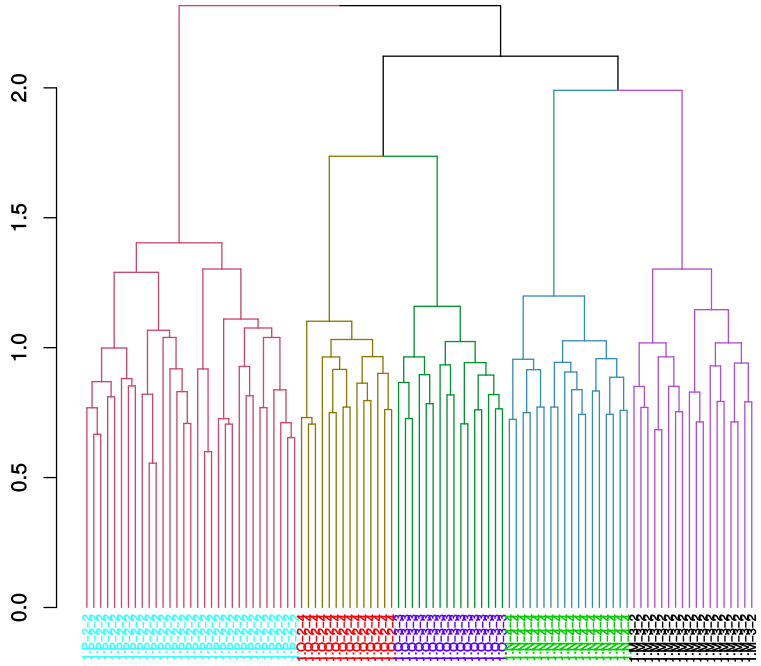


Figure 6.6: Dendogram of Subset A under the best configuration, which is achieved by all the similarity coefficients, many bin widths and method ward.D2. The colours on the dendogram depict the 5 resulting clusters after cutting the tree. Below, the samples assigned to each cluster are coloured and named after their actual book block, their actual class. The dendogram shows a perfect clustering for Subset A.

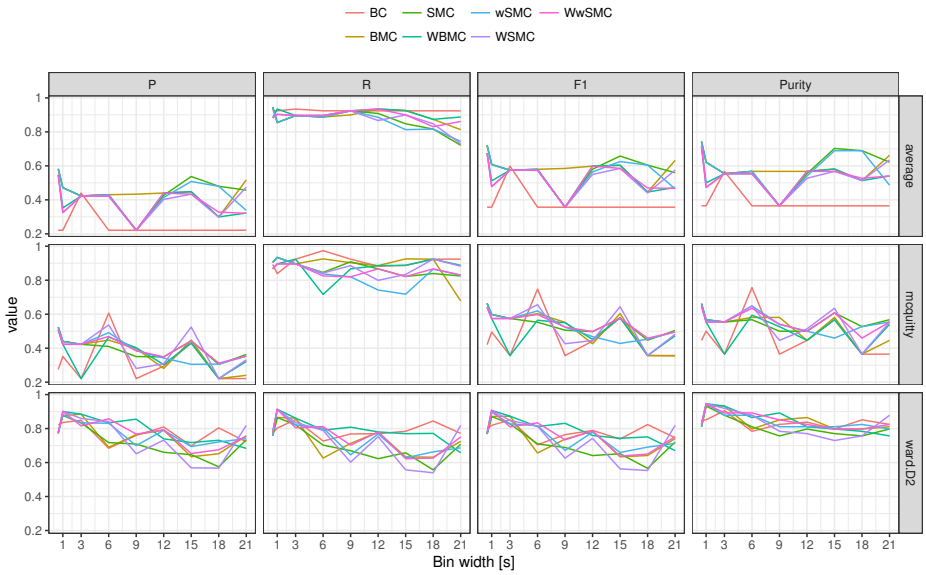


Figure 6.7: Results of clustering subset B book blocks (classes).

Results of clustering Subset 50 using the measures and methods proposed are shown in Figure 6.9. The increase in number of blocks involves an overall decrease of clustering performance. The best figures are $Purity = 0.82$ and $F1 - score = 0.75$, achieved with *ward.D2* method, BMC similarity coefficient and a bin width of 12 seconds. This result reveals that the method does not scale up in number of samples.

For a better understanding, the analysis is extended by reconsidering the block membership. Now, books that belong to an adjacent block to the block assigned by the clustering are counted as correctly classified. Subset B blocks, depicted in Figure 6.4 are an example of such condition. A block membership is extended to consider as members books placed 1-block-away along the shelf: right; left; up; or down. Figure 6.10 shows the results of extending the block membership. Following the relaxed membership condition, the best figures become $Purity = 0.91$ and $F1 - score = 0.80$. There is a clear increase of performance, meaning that a significant share of clustering error is due to a 1-block-away mistake.

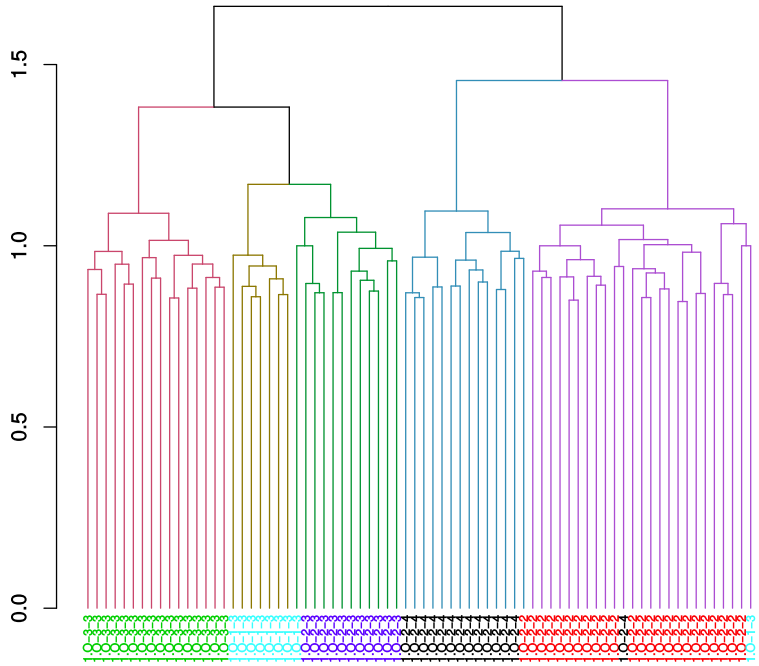


Figure 6.8: Dendogram of Subset B under the best configuration: similarity coefficient WWSMC; method ward.D2; and bin width 1 second. The colours on the dendogram depict the 5 resulting clusters after cutting the tree. Below, the samples assigned to each cluster are coloured and named after their actual book block, their actual class. There are 4 samples that do not coincide with the majority within the class (colour mismatch). Anyway, the dendogram shows a good clustering, 70 out of 74 samples are assigned to the correct class, for Subset B.

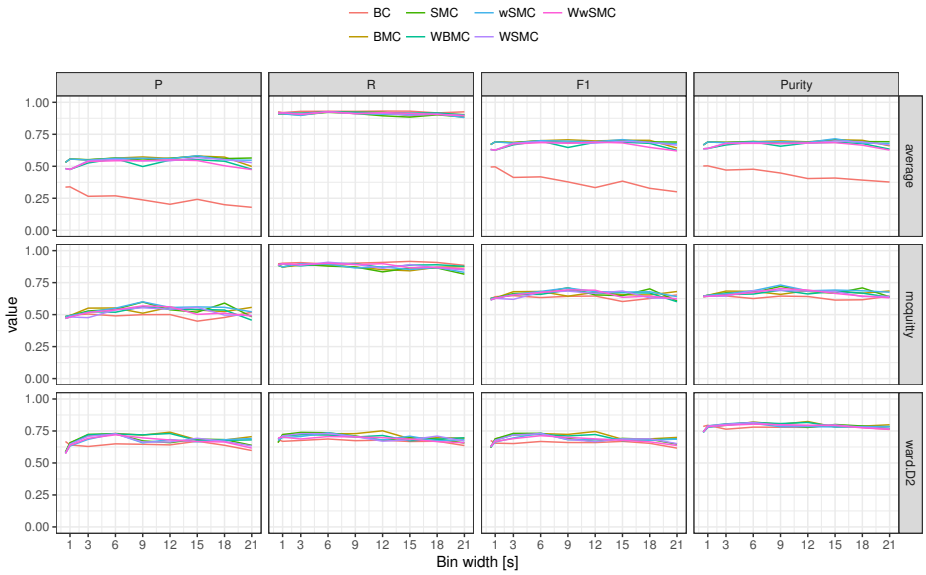


Figure 6.9: Results of clustering Subset 50 book blocks (classes).

Again, the best method is $ward.D2$, at a bin width of 9 seconds and SMC similarity.

An additional relaxation of the evaluation condition is considering books that belong to adjacent blocks across the shelf rack as members of any of the adjacent blocks. These are mirrored blocks, the ones on the other side of the same shelf rack. Same shelf rack, same block position, but facing opposite directions. This is, reachable from neighbouring corridors. Actually, adding this condition to the former, the 1-block-away 2D relaxation is extended from blocks in a 2-D plane to blocks in the three directions (1-block-away 3D). In the case of the library this may be of significant importance because there is no actual separation between the books in the two different sides of a shelf rack. This means that identifications are likely to happen simultaneously for both blocks, given books are adjacent. Figure 6.10 depicts the results. In this case, the best performance is given by the $average$ method, offering a $Purity = 0.97$ and $F1 - score = 0.96$ with WBMC similarity and a bin width of 15

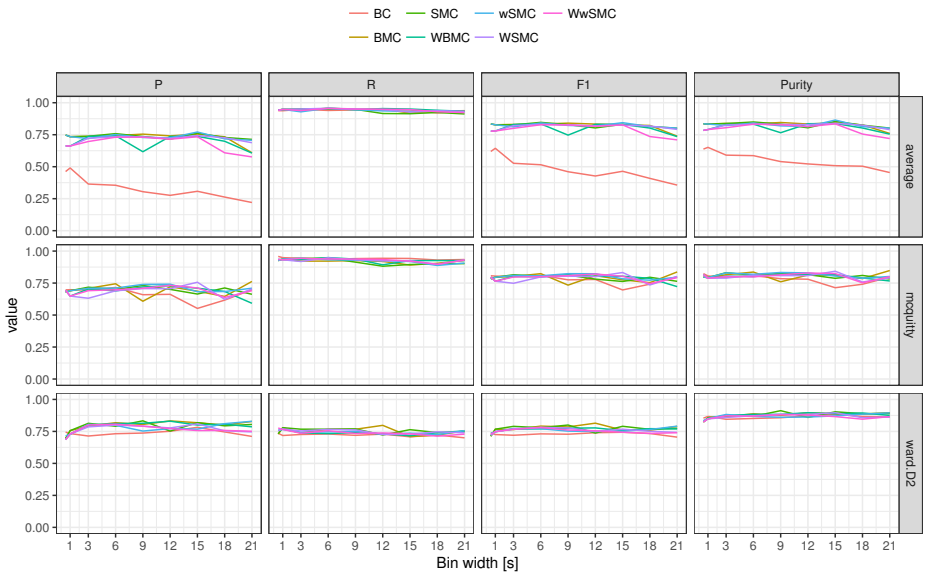


Figure 6.10: Results of clustering Subset 50 book blocks (classes) considering samples assigned to a block contiguous, along the shelf, to their actual block as correctly classified.

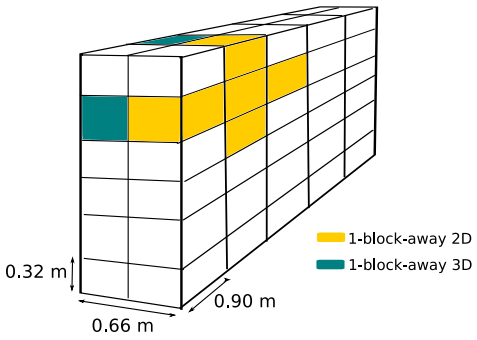


Figure 6.11: Results of clustering Subset 50 book blocks (classes) considering samples assigned to a block contiguous, along the shelf, to their actual block as correctly classified.

seconds. Methods $ward.D2$ and $average$ show similar figures but the $average$ method provides a higher recall thus a higher $F1 - score$. In conclusion, although the clustering is penalised after an increase in number of books and blocks, the relaxation of the membership condition to a 1-block-away 2D in the three directions (1-block-away 3D) offers very good results. In the case of the library this means clustering identifications within a volume as the one depicted in Figure 6.11, which is considered sufficient to detect significantly misplaced items.

Note that the library encompasses several shelf racks distributed on a large surface. Hence, books that are placed racks away from their theoretical position can be consistently detected. However, books that are placed a block away within the same rack from their expected location cannot be spotted with this method.

Subset All (256 blocks)

Finally, clustering is applied all RFID labelled books in the library in order to confirm the former findings. The complete set of books encompass 256 blocks and nearly 7,000 books. The evaluation of clustering considers the 1-block-away 3D membership. Results show that while the clustering performance is slightly affected by the increase in the number of books, it

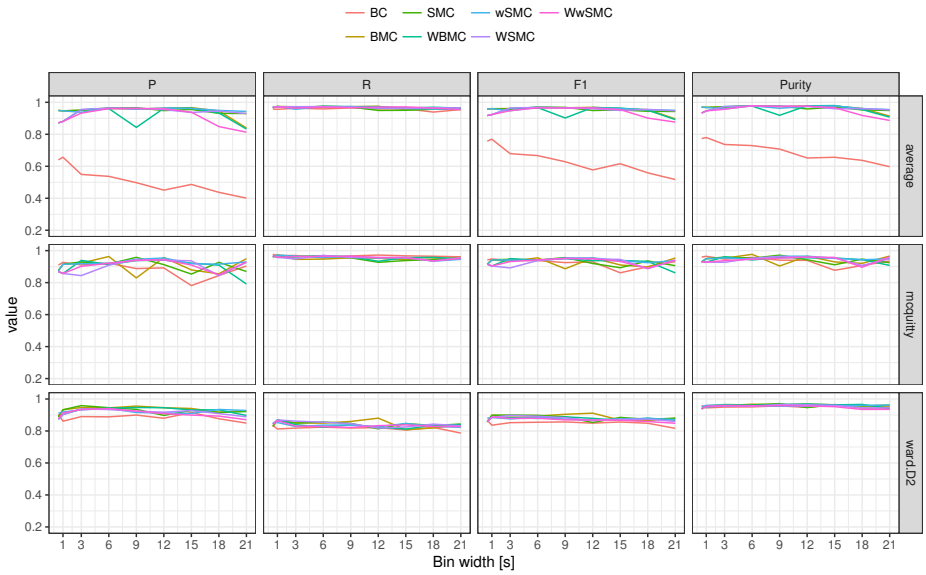


Figure 6.12: Results of clustering subset 50 book blocks (classes) considering samples assigned to a block contiguous, along or across the shelf, to their actual block as correctly classified.

offers good results: $Purity = 0.95$ and $F1 - score = 0.93$. The best performance is given by the $average$ method, SMC similarity coefficient and a bin width of 9 seconds. Figure 6.13 shows the results, which confirm that the $average$ method outperforms $ward.D2$ and $mcquitty$ at applying the 1-block-away 3D membership extension.

6.2.4 Discussion

Clustering streams of identifications as a location approach unlocks two different applications. The first relies on the use of reference objects. The locations are assigned, after clustering, as that of the reference object within the group. Hence, locations are discrete and assigned by proximity or similarity to the known reference object location. A possible downward of such solution in a real scenario is the lack of control on the placement

	max F1	max Purity	Method	Similarity	Bin width	Blocks	Membership extension
Subset A	1.0	1.0	ward.D2	All*	Many*	5	-
Subset B	0.91	0.95	ward.D2	WBMC WSMC WwSMC	1000	5	-
Subset 50	0.75	0.82	ward.D2	BMC	12000	50	-
Subset 50	0.80	0.91	ward.D2	SMC	9000	50	1-block-away 2D
Subset 50	0.96	0.97	average	WBMC	15000	50	1-block-away 3D
Subset All	0.53	0.68	ward.D2	WSMC	6000	256	-
Subset All	0.66	0.81	ward.D2	WBMC	12000	256	1-block-away 2D
Subset All	0.93	0.95	average	SMC	9000	256	1-block-away 3D

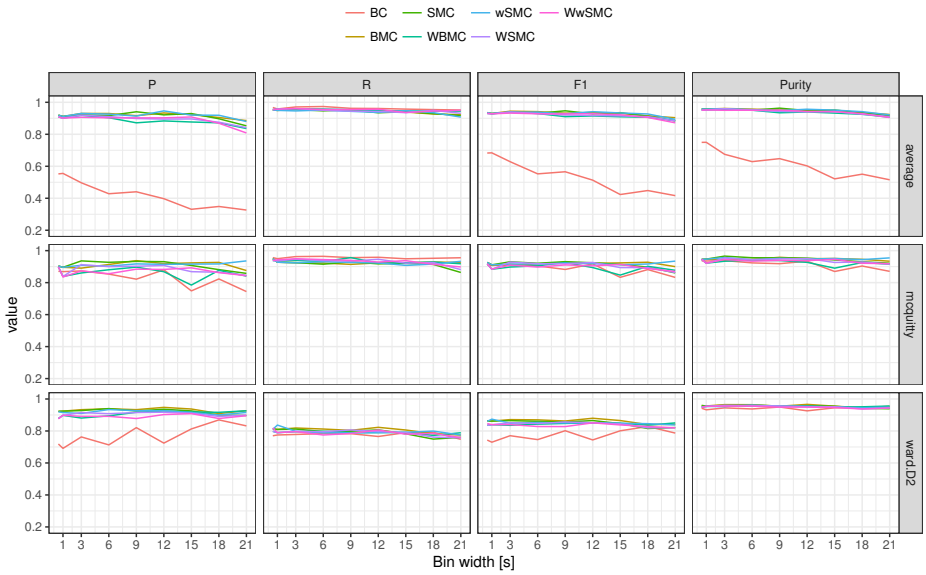


Figure 6.13: Results of clustering all the books in the library, considering samples assigned to a block contiguous, along or across the shelf, to their actual block as correctly classified.

of reference objects. Thinking of a library, it is difficult guaranteeing that a given reference book stays in the location where it was originally placed. Any time, a person could withdraw the book for consultation and leave it back in a different location. Given the amount of interactions that happen in a library, this risk is high. One should note that for a reference object to be valid, in the case of the library it should necessarily be a book. It cannot be an RFID label attached to a shelf since the detection characteristics would not match that of the books and the clustering may fail. Hence, if the target scenario for location cannot guarantee a fixed location of the reference objects, it may not be feasible. In that cases, the second possible application of clustering can be considered.

The second location application to clustering can be considered a pseudo-location solution. Not having reference objects, one cannot infer the location of the resulting clusters. However, one could still detect

misplaced objects by distinguishing the type of object in each cluster. The objects in a cluster that do not belong to the majority type are potentially misplaced. Noteworthy, RFID coding standards contemplate storing the type of object within the Electronic Product Code (EPC). Thus, the type of each object in a cluster can be determined. Getting back to the library case, the books within a computed cluster allegedly belong to a block of books, which should have, if properly ordered, consecutive signatures. Moreover, books in a block belong to a same category of books, for instance Machine Learning. Consequently, any signature or category in a given cluster that does not match the majority, is potentially misplaced. In practice, such check may not be direct, in the particular case of the library is not. Anyway, the EPC stores the book unique identifier, a barcode that can be used to query the library database for the book's signature and category. This pseudo-location approach can be valid as well in retail stores. For instance, clothes are usually placed in a store grouped by type of garment, brand and/or size. Such product features can be used to discern whether an object in a cluster belongs to the family determined by the majority. For instance, a piece of underwear found among suits is likely to be misplaced.

6.3 Recursive Bayesian estimation

The second approach to the location of items is recursive Bayesian Estimation [54], or Bayes filter. In brief, it is the recursive application of Bayes rule to determine the most likely location of an item given a sequence of observations. Let $L_{epc} = \{l_a, l_b, ..., l_q\}$ be the state vector, the collection of all possible locations of an arbitrary item epc, where $l_a = (x_a, y_a, z_a)$ is a three-dimensional position around the robot. Let $ID_{epc} = \{id_1, ..., id_k\}$ be the sequence of identifications of item epc and $R_{robot} = \{r_1, ..., r_k\}$ the corresponding robot poses, where $r_i = (x_i, y_i, \theta_i)$. The sequences of identifications and robot poses carry implicitly the timestamp, which is used to match them.

Following Bayes rule and assuming that an identification is indepen-

dent of the former, the recursive update formula can be expressed as follows.

$$\underbrace{p(l_a|id_{1:k}, r_{1:k})}_{posterior} = \eta \cdot \underbrace{p(id_k|r_k, l_a)}_{\text{measurement likelihood}} \cdot \underbrace{p(l_a|id_{1:k-1}, r_{1:k-1})}_{\text{prior}} \tag{6.24}$$

The posterior probability of an arbitrary location l_a given a sequence of measurements and robot poses is proportional to the likelihood of the measurement and the prior knowledge about that location, computed from previous observations. The factor η normalises the probability among all other possible locations within the vector of states L_{epc}.

The measurement likelihood expresses the probability of an identification given a robot position and a possible item location. In order to compute the likelihood, an identification model is required. In nature, it is environment-dependant. In practice, it is not feasible mapping each pair of robot positions and item locations. Furthermore, such model would be tightly bound to the environment's configuration, which implies not being applicable to a similar space. For this reasons, the likelihood is transformed to rely on the relative position between the antenna and the item location. It is expressed as

$$p(id_k|r_k, l_a) \longrightarrow p(id_k|\delta_{ant_k}(r_k, l_a)) \tag{6.25}$$

where $\delta_{ant_k}(r_k, l_a)$ is the relative position at a kth identification.

6.3.1 Identification model

The identification model provides the information needed to compute the likelihood of an identification. In this case, each identification is considered as $id_i = \{ant_i, rssi_i\}$, which implies taking into account only the antenna ant_i and the received signal strength $rssi_i$ from the information reported by the reader, expressed in Equation 6.2. Evidence from prior works [55, 36, 37, 38] that approach location with success using only

$rssi$ is the justification to discard other RFID parameters. The identification model is computed antenna-wise. This is due to the fact that in the library, books are placed at discrete heights, resulting the detection pattern of each antenna notably different along the z axis. Hence, the likelihood can be reformulated as

$$p(id_k|r_k, l_a) = p(rssi_k|\delta_{ant_k}, ant_k) \tag{6.26}$$

and the identification model M is the collection of $rssi$ probability density functions at all relative antenna positions. Each relative position has its own $rssi$ probability density function, which is used to compute the likelihood of a measurement at that position.

$$M = \{\ f_{RSSI}(rssi|\delta_{ant_i}) \quad \forall ant_i\} \tag{6.27}$$

The identification model is learned in a supervised fashion. A subset of the location dataset is used as training data. Specifically, 4 books from each of the six shelves heights are selected, a total of 24 books. Note that, in reality, these are reference items whose locations should be measured in purpose to learn the identification model in a real case. In this sense, the fewer the better. As a consequence, training data is expected to be incomplete.

The relative positions of the identifications at training are conditioned by the robot navigation. In most cases, at inventorying, the robot follows a path that does not covers all the possible relative locations between the antennas and the reference objects. Logically, the sparsity of data is more notable the fewer the reference objects and the shorter the training path. In a real scenario, such limitation is implicit as a result of the structural distribution of the layout and a limited time for training. As an example, let's take the case of an apparel retail store with 12 reference objects at representative locations. The robot is commanded manually with the aim of collecting the training set. However, the robot can only navigate where there are no obstacles, which hampers the objective of registering identifications at all relative object to robot locations. Virtually, it is impossible

to place enough reference items or follow a path that can cover all relative locations. One could reason that by placing just one reference object surrounded by free space that should be possible. Yet, identifications are conditioned by the environment where objects are placed. Essentially, the identifications of books placed in shelves next to other books or piled clothes are not the same as if they would be isolated. For this reason, the training data is necessarily collected at the target scenario.

The detection model is computed by partitioning the space and fitting a probability distribution to the samples in each of the resulting bins. In practice, the identification model consists of a 3D grid. Each voxel encloses two parameters, the mean and standard deviation of the estimated probability distribution. A third property is implicit: the number of samples. Due to the sparsity of data, bins containing no samples cannot be considered to represent a null probability. Furthermore, there are bins that do not contain enough samples to estimate a probability distribution. Consequently, at computing the detection model, the probability of a number of bins needs to be inferred. For that, an imputation algorithm is proposed.

Estimation of the $rssi$ density function

Prior works have applied Bayesian methods to the location of items assuming a Gaussian distribution of the received signal strength [55, 36, 37, 38]. We presume a Gaussian distribution can be measured in ideal conditions, mainly for line-of-sight identifications. However, this is not the case in the library. Moreover, prior works encompass less than 100 items while the library encloses 7,000. In the library case, the interference between books is expected to distort significantly identifications compared to a controlled environment. For all this, the complete dataset collected at the University Library is analysed in order to determine the signal strength distribution. The Cullen and Frey plot provides a simple means to analyse the distribution of a set of samples. It is based on kurtosis and skewness, which are properties of the shape of a probability distribution. Kurtosis measures the distribution's tails shape and skewness measures a distribu-

tion’s asymmetry. The Cullen and Frey plot provides a visual comparison of kurtosis and skewness with those of known parametric distributions.

First, the distribution is analysed for all the identifications within an inventory round (inv_5 from Table 6.1). The sum of normal distributions, if the random variables are independent, is a normal distribution. Figure 6.14 shows that the $rssi$ distribution is not near to the Gaussian distribution. The $rssi$ distribution falls close to the dashed line that marks the gamma distribution. Apparently, the $rssi$ density function is not Gaussian.

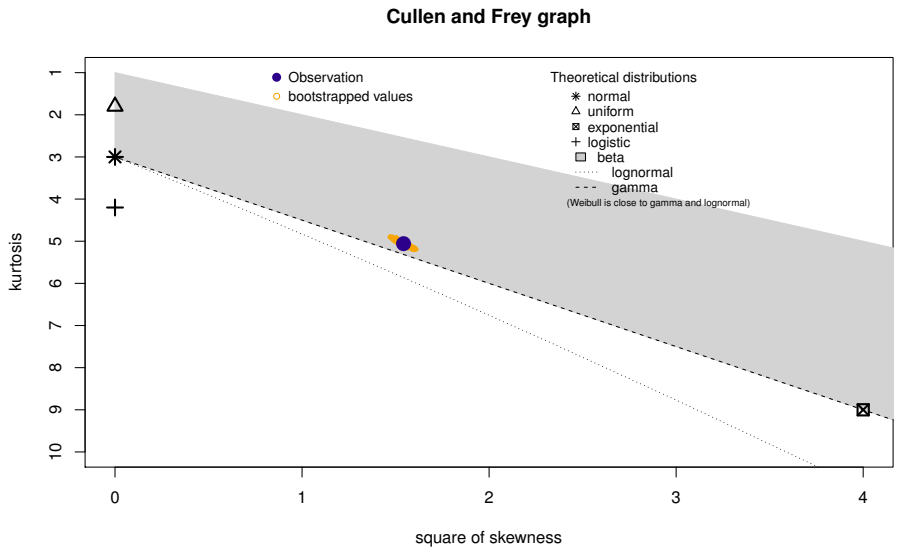


Figure 6.14: Cullen and Frey graph of the samples of inventory round inv_5. The plot shows that the probability distribution of the identifications is not close to a Gaussian.

Next, the distribution is analysed for each voxel individually. The aim is understanding if the the distribution is the same for all the relative antenna-item locations. Only voxels enclosing more than 20 samples are considered to discard voxels not statistically representative. Figure 6.15 shows the density of voxel distributions falling on the different regions of the Cullen and Frey graph. The Gaussian distribution region of the graph

encloses only 15% of voxels. The remaining are spread across the rest of the regions. In conclusion, the $rssi$ density function cannot be generally considered Gaussian for the library dataset. Apparently, the density function depends on the relative antenna-item location. This is unsurprising since measures are taken at a real world scenario as opposed to a controlled environment, where a Gaussian distribution is more likely to be measured. In any case, given a majority of the bins show a Gaussian distribution, this is used for the computation of locations. Yet, location is computed as well using a Uniform distribution. Essentially, using a Uniform distribution that spans from the global minimum $rssi$ to the global maximum $rssi$, the $rssi$ is implicitly ignored. The reason for that is the apparent impossibility to characterise a density function that applies to all the voxels.

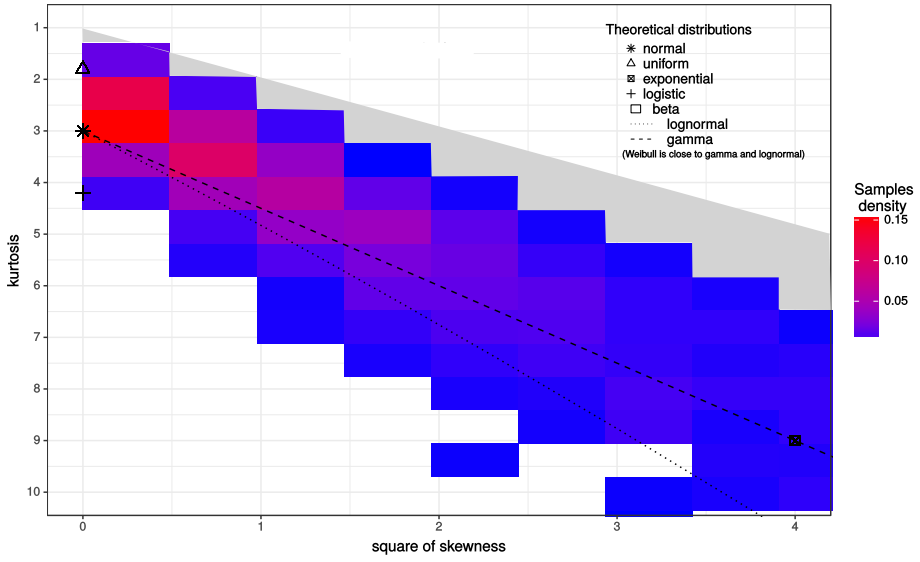


Figure 6.15: Cullen and Frey graph of the samples of inventory round inv_5 for each relative antenna-item location. The plot show the density of points falling on each region. There is not a distribution that characterises univocally the different antenna-item locations.

Imputation of missing data

The imputation of missing data aims at filling the model at relative antenna-item locations where not enough data is available for a statistically significant measure. This problem was tackled by Vorst et al. [55] following a K-nearest neighbours approach, which is extended in this thesis. The main idea is constraining the K-nearest neighbours to a maximum distance D_{max} from the voxel whose $rssi$ mean and standard deviation has to be determined. In this sense, if within the maximum distance there are K or more identifications, the mean and standard deviations of the $rssi$ are assigned to the voxel in scope. Otherwise, the voxel is considered empty. The reason for that is avoiding the imputation with far away identifications of voxels that have been explicitly observed empty.

The space is strategically partitioned forming an octree (a 3-dimensional tree), which implicitly means that each node has eight siblings. The imputation algorithm, Algorithm 3, first checks the amount of identifications in the leaf node. If they are equal or greater than K, the leaf is assigned these identifications $rssi$ mean and standard deviation. Otherwise, the node's siblings identifications are taken in consideration. Again, if these do not encompass enough identifications, the algorithm moves to the parent node and checks again the amount of identifications at that level. The operation is repeated, moving up the tree until K identifications are found or until the maximum distance D_{max} from the node is reached. The values of K and D_{max} are determined empirically.

Algorithm 3 Imputation algorithm

```
1: for leaf in tree do
2:     node ← leaf
3:     identifications ← ∅
4:     while length(samples) < K do
5:         if d(node, leaf) ≤ D_max then
6:             identifications ← getIdentifications(node)
7:             node ← parent(node)
8:         else
9:             identifications ← ∅
10:            break
```

K: minimum number of samples
D_{max}: maximum expansion distance

In order to illustrate the imputation, Figure 6.16 depicts identifications and corresponding models for the training set, which consists of 24 books, 4 books at each shelf height. The training set is limited and, in consequence, the sparsity of identifications significant. For visualisation purposes a single antenna (L1) and a single relative height (-11 cm) are shown. Hence, the models shown are a slice of the complete identification model. The parameters chosen and combined are: $K = (1, 5, 10)$ samples; and $D_{max} = (60, 120, 180)$ cm. On one hand, a group of computed models show an expected over-imputation. For instance the pair $D_{max} = 180$ and $K = 1$. This is due to a very relaxed condition for considering a set of samples statistically significant ($K = 1$). On the other hand, there are computed models that suffer from under-imputation. For instance $D_{max} = 60$ and $K = 10$. In this case due to a limited extent in the distance ($D_{max} = 60$) for gathering a considerable number of samples ($K = 10$). Nevertheless, there are models that apparently fit the samples. For instance ($D_{max} = 180$, $K = 10$), ($D_{max} = 180$, $K = 5$) and ($D_{max} = 120$, $K = 5$). The performance of the resulting models is analysed experimentally by computing the locations of the books in the dataset.

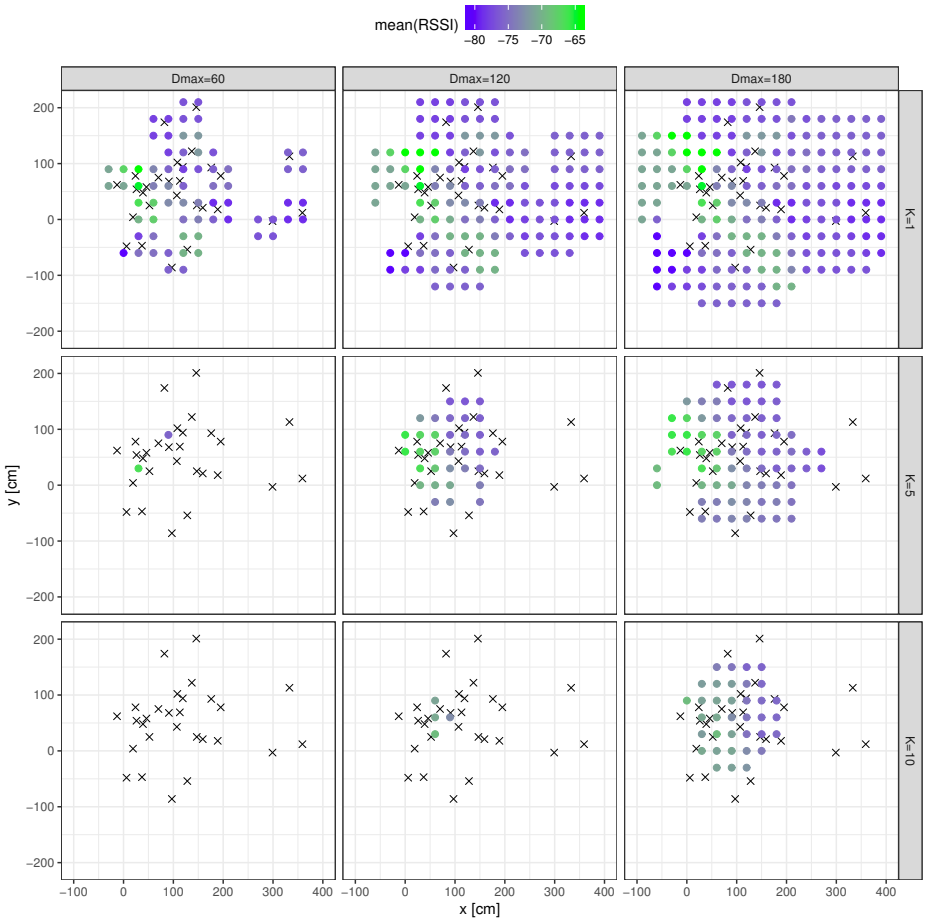


Figure 6.16: Identification models computed with the combination of $K = \{1, 5, 10\}$ and $D_{max} = \{60, 120, 180\}$ for a single antenna and height. The black crosses mark identifications, which can overlap. The coloured dots represent the $rssi$ mean of the computed identification model. The models depicted grid resolution is 30 cm. A 2-D slice (one height) of the complete 3-D model is shown for visualization purposes.

6.3.2 Experimentation

The assessment of the Bayesian location approach is undertaken using the library dataset. The stages for applying the location algorithm are three:

running a training round; computing the model; and computing locations.

Training round

The model is computed from a training set that includes 24 books, 4 books per shelf height. The training set corresponds to a manual inventory round taken with the aim of capturing a high number of different relative antenna to tag poses. Consequently, the robot was guided to do several turns and passes. An important consideration is that, generally, the training set will not consist of a high number of references. This is due to the fact that in a real scenario, reference objects need to be placed and located manually. Therefore, the less reference objects, the most convenient. Yet, they have to be representative of the complete set of objects. In conclusion, the training round aims at capturing a high number of identifications from several antenna to tag relative positions using a reduced number of reference objects.

Model computation

Observations from the training round are used to compute a set of detection models under different configurations. In this manner, the optimal combination of parameters N_{max} and D_{max} can be determined empirically. The parameters analysed are combinations of: $N_{max} = (1, 5, 10)$ samples; and $D_{max} = (60, 120, 180)$ cm. Additionally, given there is no conclusive evidence regarding the probability distribution of the samples, the model is considered in two flavours: Gaussian and Uniform distribution of samples. The model variables, mean and standard deviation, do not vary but it does the computation of the observations likelihood.

Location computation

Locations are computed using observations from a usual autonomous inventory round, ignoring the detections of books that belong to the training set. The inventory round is faithful to the navigation behaviour determined in Chapter 5 for inventory, it is not modified for location purposes.

A random subset of 1000 tags is selected for the analysis of computed locations.

Results

The results of applying Bayesian Recursive Estimation to the location of items are shown in Figure 6.17 for the case considering the RSSI probability distribution of the model as uniform. Given an identification, and its corresponding RSSI, the likelihood of the observation is the same for each position of the model. Results evince the relevance of computing properly the model. Best results are yield by the combinations ($D_{max} = 120$, $N_{min} = 5$) and ($D_{max} = 180$, $N_{min} = 10$). Such combinations of parameters are coherent with the imputation initial hypothesis. Samples need to be gathered from upper positions in the tree (non-restrictive D_{max}), for a statistically significant model.

A likelihood modelled with Gaussian probability distribution is analysed as well, for comparison. Figure 6.18 shows the results, which are in general worse than the uniform distribution results. Surprisingly, there are combination of parameters that fail completely at location. This confirms the conclusion of the probability distributions analysis. The RSSI is not Gaussian, at least in the library scenario. Forcing the likelihood as Gaussian misleads the updates of the Bayesian estimation. On the other hand, a Uniform distribution outputs better results. Considering the RSSI to be distributed uniformly is equivalent to wiping it out from the computation and keeping only the part of the model that is purely spatial. The latter makes sense since in a real scenario, RSSI is very affected by the configuration of the surrounding space. In conclusion, in this case the RSSI cannot be modelled, thus it does not provide information for updating the estimation. Table 6.7 presents comparative numeric results, focusing on the combinations of parameters that yield best results, both for the Gaussian and Uniform cases.

An additional analysis of results focuses on the contribution of identification instances (per object) to the location algorithm. The question to answer is: Do more identification instances (samples) contribute to a bet-

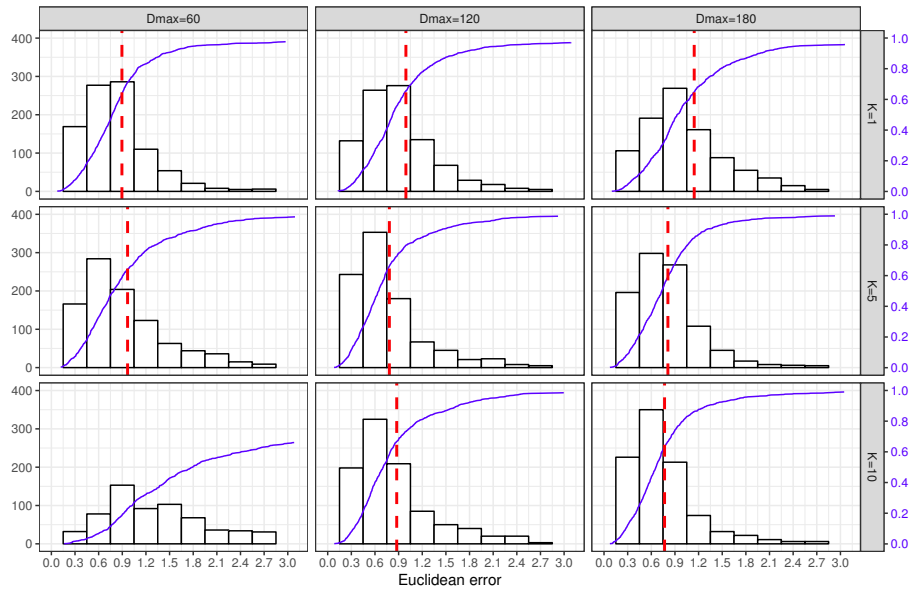


Figure 6.17: Histograms of the 3-D location euclidean error for the different combinations of model parameters considering a Uniform RSSI probability distribution. The grid is composed of: horizontally $D_{max} = (60, 120, 180)$; and vertically $N_{max} = (1, 5, 10)$ samples. The red dashed line marks the error mean and the blue line depicts the cumulative histogram. The lowest error is given by the combinations ($D_{max} = 120$, $N_{min} = 5$) and ($D_{max} = 180$, $N_{min} = 10$).

ter location of the given object? This is analysed for two reasons. First, the more samples, the longer algorithm computation. Second, given a high number of samples, some could be far identifications, thus noise for the algorithm. Figure 6.19 depicts the relation between the 3-dimensional euclidean error and the number of samples. Each point represents a single object and the loess regression is computed. The euclidean error shows an initial decreasing trend with a minimum at 10 samples per object. After, it increases with the increase in number of samples. This can be explained by the fact that the higher the number of identifications, the higher diversity of robot positions that originated them. Figure 6.20 evinces that the more samples, the higher standard deviation of robot positions. Hence, the error increases with the increase of positions originating the identifi-

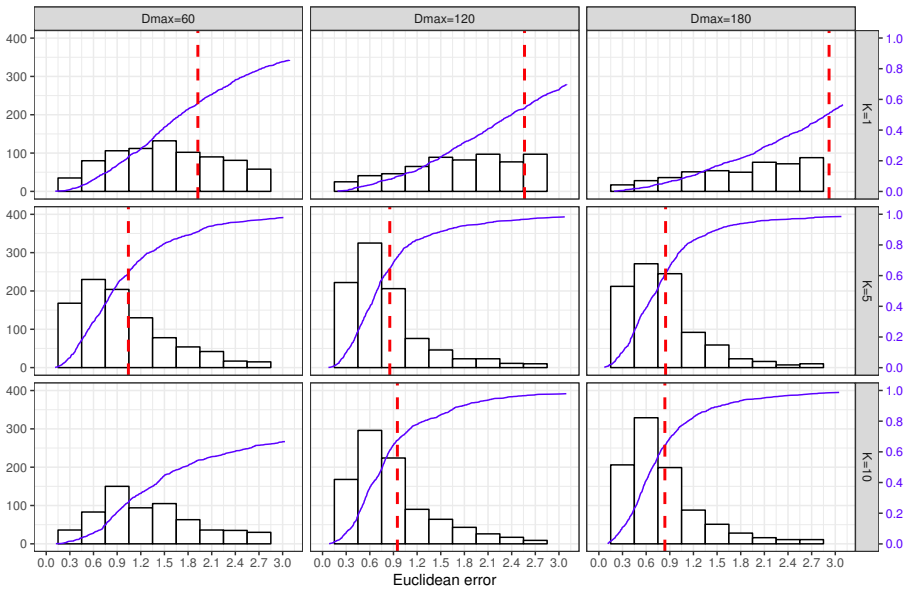


Figure 6.18: Histograms of the 3-D location euclidean error for the different combinations of model parameters considering a Gaussian RSSI probability distribution. The grid is composed of: horizontally $D_{max} = (60, 120, 180)$; and vertically $N_{max} = (1, 5, 10)$ samples. The red dashed line marks the error mean and the blue line depicts the cumulative histogram. The lowest error is given by the combinations ($D_{max} = 120$, $N_{min} = 5$) and ($D_{max} = 180$, $N_{min} = 10$).

cation of an object. The most plausible explanation is that while a subset of close identifications, those high in RSSI, contribute positively to location, far identifications penalise it. In order to confirm such hypothesis, the location algorithm is run taking into account only the 10 and 25 highest RSSI samples. Table 6.7 shows the results. The error is considerably reduced by using only the 10 highest RSSI samples. The most notable reduction in the location error is at the highest spectrum of error. The 95% quartile of the euclidean error is reduced from 1.72 cm to 1.32 cm by limiting the identification instances to the 10 samples with a highest RSSI. And the benefit of limiting the amount of samples is not only a better performance of the location algorithm but also a shorter computation time.

Table 6.7: Summary of location results for the different combination of parameters analysed. For simplicity, it includes the most significant combinations of parameters, the ones that offer a better performance in their group of analysis. Reducing the number of detection instances in the computation of locations, reduces significantly the error. The minimum error is given by using the 10 identifications instances with the highest RSSI.

Model parameters			Computation parameters	3-D location euclidean error					
N_{min} [samples]	D_{max} [cm]	Probability distribution	Detection instances	median [cm]	mean [cm]	std	q75 [cm]	q90 [cm]	q95 [cm]
10	180	Gaussian	All	0.68	0.84	0.60	1.00	1.55	2.07
5	120	Gaussian	All	0.69	0.85	0.64	1.02	1.51	1.97
10	180	Uniform	All	0.66	0.77	0.54	0.91	1.30	1.72
5	120	Uniform	All	0.65	0.78	0.57	0.92	1.45	1.94
10	180	Uniform	25	0.61	0.72	0.47	0.88	1.23	1.57
5	120	Uniform	25	0.59	0.71	0.49	0.84	1.21	1.58
10	180	Uniform	10	0.62	0.69	0.42	0.83	1.08	1.38
5	120	Uniform	10	0.61	0.68	0.44	0.81	1.10	1.32
10	180	Uniform	5	0.81	0.87	0.47	1.09	1.39	1.67
5	120	Uniform	5	0.79	0.89	0.53	1.11	1.47	1.86

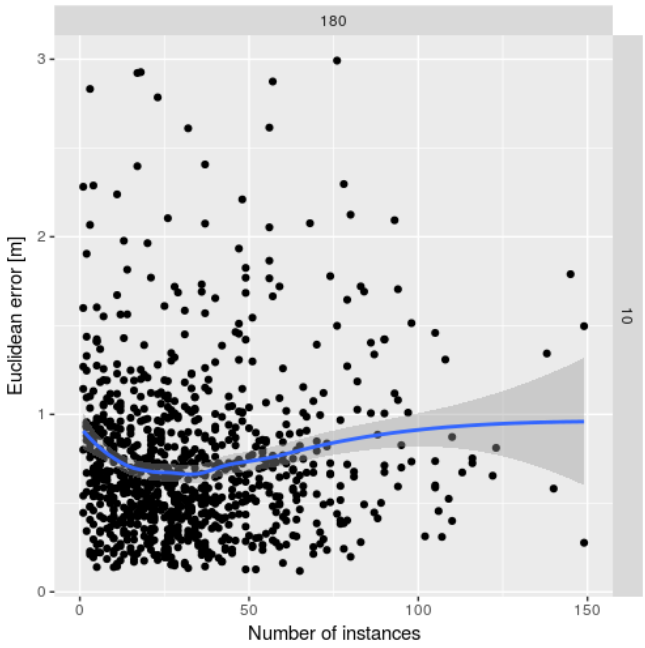


Figure 6.19: Relation between the euclidean error of the location and the number of identification instances for each object detected. One point of the plot represents one object. In blue, the loess regression of the samples. There is a minimum around 10 instances per object. The plot shown corresponds to model parameters ($D_{max} = 180$, $N_{min} = 10$) and a uniform RSSI distribution.

6.3.3 Conclusion

Bayesian Recursive Estimation is proved a valuable method for RFID location. A proper model computation is of utmost importance and it requires imputation given the amount of training data is likely to be scarce in real scenarios. Importantly, the probability distribution of the RSSI needs to be carefully analysed. While in controlled environments it can be measured a Gaussian distribution, this is not the case in the library. Additionally, the best locations have been measured by taking only the 10 highest RSSI identifications of each book. The best computed locations show a mean error of 0.68cm and a 95% quartile at 1.32cm.

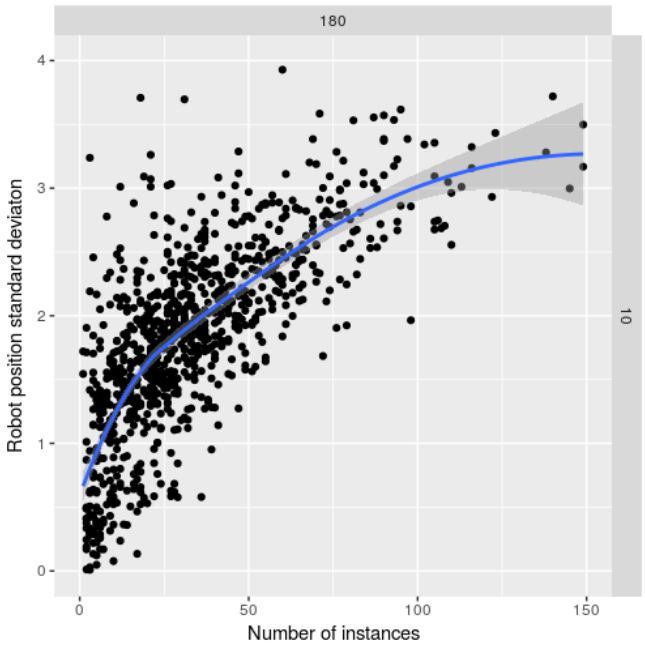


Figure 6.20: Relation between the aggregated standard deviation of robot positions that originate detection instances and number of instances for each object detected. One point of the plot represents one object. The higher the number of instances, the more diversity in robot positions. The plot shown corresponds to model parameters ($D_{max} = 180$, $N_{min} = 10$) and a uniform RSSI distribution.

Chapter 7

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

A solution for the automation of inventory and location of items that combines Radio Frequency Identification (RFID) and mobile robotics has been proposed and validated in real environments. The navigation of the robot is necessarily commanded by the progress of RFID identifications in order to deliver an accuracy higher than 99.5%. An algorithm that takes as input the progress of RFID identifications and commands the robot to stop and twist in place and resume the navigation has been implemented. Results show that adjusting the algorithm parameters is critical for the robot efficiency in terms of not only inventory accuracy but duration of as well. In this sense, the robot is demonstrated to be more accurate, precise and faster than a person with a handheld RFID reader, the state-of-the-art solution used for inventorying. In order to assess the comparative performance of the robot and a handheld, a methodology, which includes specific measures has been proposed. To this thesis date, a similar analysis had not been approached thoroughly. The assessment required as well of a controlled environment were to undertake tests. With this purpose, an evaluation environment was set up in the Pompeu Fabra Library, enclos-

ing 7,000 RFID labelled books. Validation tests were performed as well at an actual retail store. Robot inventory results were compared to handheld inventories taken by associates. Results showed that the robot is more accurate, more precise and faster approaching the complex zones. However, the person was faster and showed equivalent accuracy and precision figures in the simplest zone. The robot paths are longer due to rerouting in front of unexpected obstacles, while a person can simply jump over them. On the other hand, a person's capacity to plan efficiently an inventory does not scale with complexity, which makes the robot better at longer inventories.

Location of items has been analysed following two different approaches. The first proposed approach is clustering identifications and assigning to each cluster the location of a known reference RFID labelled product. For that, a set of measures that quantify the distance between two streams of identifications, those of two different objects, have been proposed. The only parameters used in the computation of distance are the time of the identification and the antenna. Hierarchical clustering has been applied to subsets comprising from 5 blocks to all the products in scope in the Pompeu Fabra Library test environment. Results show that clustering using the proposed measures is feasible for a small amount of samples but does not scale up to all the samples. However, if the condition for a book to be considered correctly assigned to a cluster is relaxed to neighbour classes, the results of clustering are satisfactory using all the books in scope. With the former condition, all the books in the library can be clustered with F1=0.92 and Purity=0.95.

The second proposed location approach uses Recursive Bayesian estimation. In this case, the items location obtained refers to a map with known cartesian 3D coordinates. A detection model is computed in supervised manner placing 20 reference books in the test environment. Given the amount of reference objects is limited, it requires work hours placing and computing the location of the references, an algorithm for the imputation of missing data is proposed. The algorithm consists in combining the sparse data recorded to compute a model that is informed within an antenna's reading reach. Recursive Bayesian estimation has been com-

puted using different models obtained from the combination of imputation parameters. The best location result obtained shows a 3-D euclidean mean error of 0.68 cm and a 95-quartile of 1.32 cm. The dataset recorded and used in the computation and assessment of the location algorithm is shared openly with the community.

At the end of this work, the robot has been demonstrated valuable to retailers and is market-ready.

7.2 Future work

Future work identified involves the complete automation of robot operation. For that, an unassisted exploration of the environment will be pursued, which would save the need of running a manual recognition before inventorying. The main challenge regarding exploration is completeness and efficiency, given exploration algorithms do not scale well in large environments.

Regarding location, the algorithms proposed will be validated in real retail stores and warehouses. While results obtained are promising, the algorithms have to be validated in environments with characteristics different than those of the Pompeu Fabra Library. The clustering approach can be extended by designing a specific clustering method with the purpose of location. Bayesian estimation can be improved by preprocessing identifications and keeping only those that contribute positively to location accuracy.

A complementary application of the robot, given it carries two RGBD cameras for obstacle avoidance, is 3D mapping. The RGDB cameras observations can be used to feed a SLAM algorithm that, in combination with other sensors information, creates a 3D coloured map of the environment. With that, a synthetic view of a store or warehouse that includes architectonic features and product information can be created. Possible applications of the synthetic view are offering customers an enriched shopping experience or the virtual monitoring and management of the store facilities and products.

Appendices

Appendix A

IDENTIFICATION STREAMS

A.1 Subset A

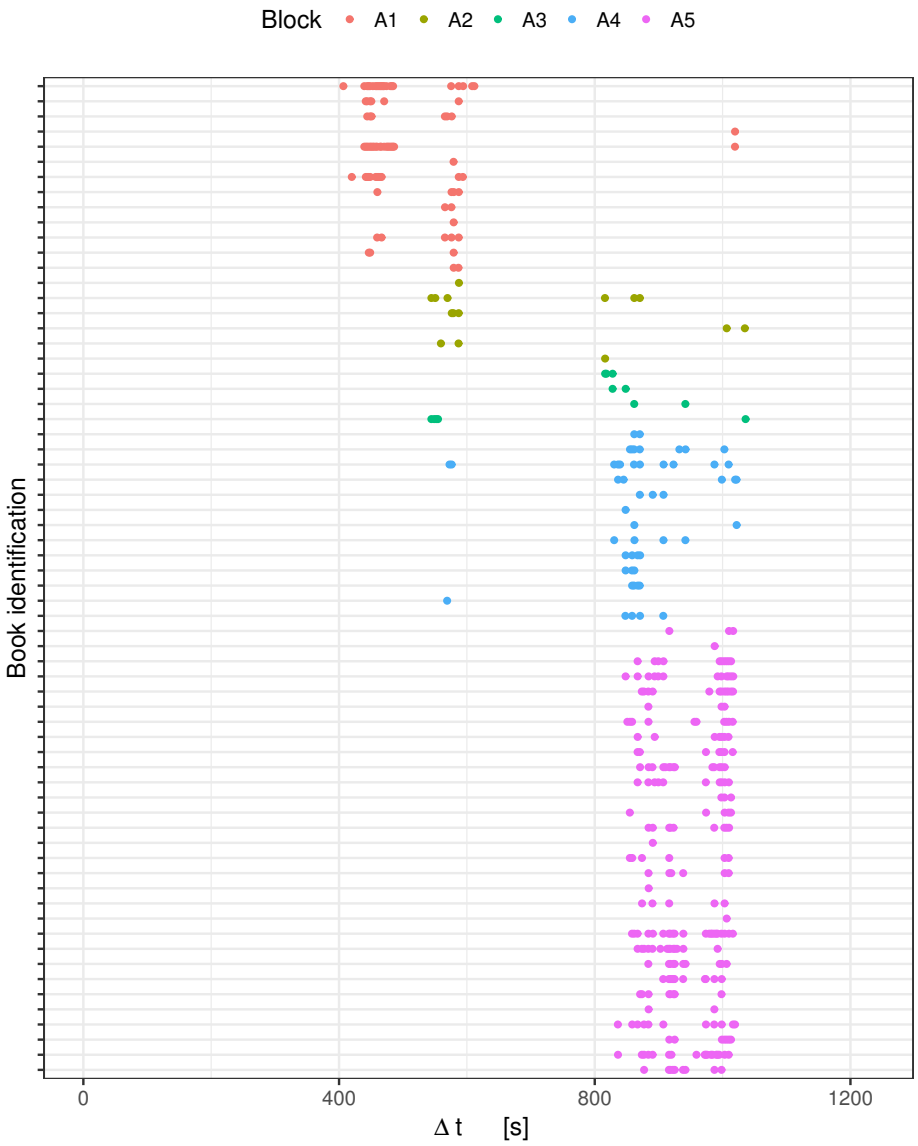


Figure A.1: Left bottom antenna identification streams of books in Subset A. Each line represents the identification instances of a single book. Blocks are grouped by colour.

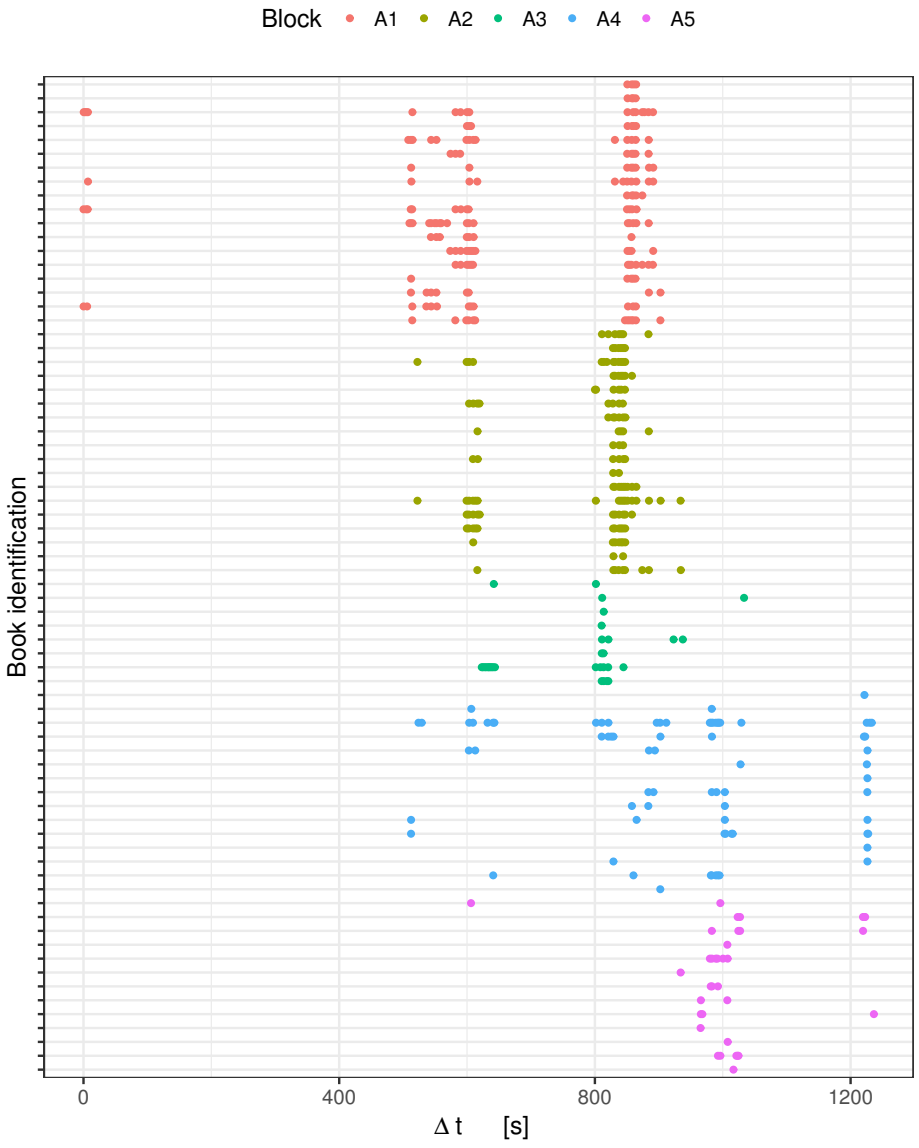


Figure A.2: Right middle antenna identification streams of books in Subset A. Each line represents the identification instances of a single book. Blocks are grouped by colour.

A.2 Subset B

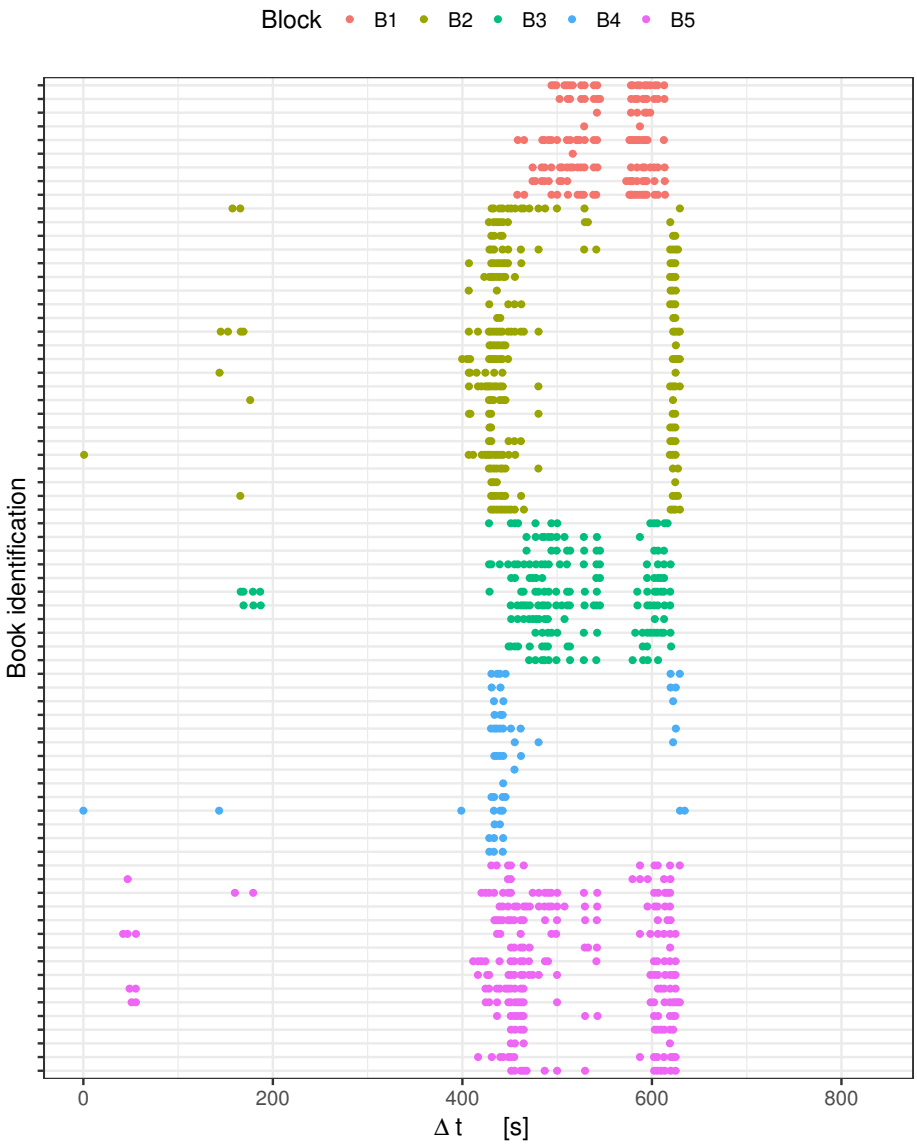


Figure A.3: Left bottom antenna identification streams of books in Subset B. Each line represents the identification instances of a single book. Blocks are grouped by colour.

Figure A.4: Right middle antenna identification streams of books in Subset B. Each line represents the identification instances of a single book. Blocks are grouped by colour.

Appendix B

3D MAP

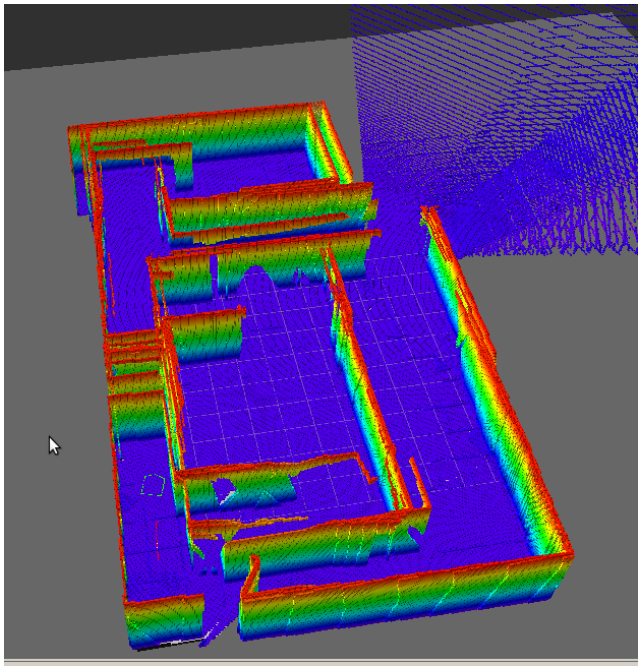


Figure B.1: View of a 3D occupancy map recorded by the robot. Colour is used as a height scale. Blue is ground level.

List of Figures

1.1 The robot taking inventory at an actual store. 2

2.1 Summary of IRI's main causes and consequences. Green coloured, the contributions of the proposed inventory robot to the mitigation of IRI. At the same time, it prevents one of the main sources of IRI and provides a means of a periodical and reliable correction. 11

3.1 (a) Proof of concept of the mobile RFID system for inventory. Labelled objects were inside the boxes on the shelves. (b) Detail of the RFID subsystem that the humanoid robot is carrying. 20
3.2 Naked mobile base used in the robot prototype. 22
3.3 Robot prototype in a store. 23

4.1 The identification subsystem consists of 12 antennas fed by 3 independent RFID readers. The identifications are gathered by a central process through an Ethernet link. . 30
4.2 RB-1 mobile base schematics. The robotic subsystem is a customization of the commercial mobile base. Source: https://www.robotnik.es 32
4.3 Laser Hokuyo UST-10LX, the one installed in the robotic subsystem. Source: http://www.hokuyo-usa.com/ 32
4.4 Orbbec Astra Mini, the RGB camera model installed in the robotic subsystem. Source: https://orbbec3d.com/ . . 34

4.5 Pixhawk Autopilot, the gyroscope/accelerometer installed in the robotic subsystem. Source: https://pixhawk.org/ . 34

4.6 Schematics of the robot design. On the left, a view of the assembled robot, with the two RGB-D cameras on top of the RFID subsystem and a side cover removed to show a group of three antennas beneath. The bottom part is the mobile base. Note the laser's 270°aperture and a front cover for a quick release of the battery. On the right, the payload is folded, a feature that was included for the ease of transportation. 37

4.7 Architecture of ROS navigation stack, the logics used by the robot for autonomous navigation. Source: http://wiki.ros.org 40

4.8 Visualization of the robot during an inventory round on RViz, the robot visualization tool provided with ROS. Several pieces of information are presented live and represent the robot's knowledge of the environment. On one hand, the occupation map and waypoints, represented by arrows, which are static data recorded during recognition. On the other hand, live sensor data: in red laser scans; in light blue the collapsed camera view. Finally, robot computations: in yellow, the local path; in blue, the global path from an actual robot pose to the next navigation goal; in purple, the navigation costmap, which sets the costs for navigation based on sensor data. 42

4.9 Store layout divided in 22 zones, following the divide and conquer strategy implemented by the robot. The triangles are the cues of the QR codes locations. The store area is 7,500 m^2. 45

4.10 The task manager takes high level task parameters from the Human-Robot interface and translates them to robot orders by following the proper sequence of states that trigger all the nodes needed during a mission. The status of the robot is monitored by the task manager and implements event-based state transitions. The mission manager connects the recognition and the inventory. Namely, it manages the sequence of zones selected during an inventory mission. Mainly, it loads the map and goals of each zone at the proper zone transitions. 47

4.11 Home page of the interface that enables the interaction with the robot. 49

4.12 Interface views at running a recognition. (a) Automatic detection of the QR that identifies a zone at its starting point. (b) During the recognition a control pad allows commanding the robot. A view of the recorded map with interest points linked by straight lines is updated continuously. In this way, the operator can consult the progress of the recognition and oversights can be prevented. (c) Automatic detection of the QR code that identifies the immediate consecutive zone, which marks the end of the current zone. In this manner, transitions between zones are guaranteed. At this position the recognition can be concluded or continued. In any case, a recognition is necessarily finished in front of the QR that marks the start of the next zone. 50

4.13 Interface views at running an inventory. (a) Automatic detection of the QR that identifies a zone at its starting point. (b) Selection of zones to be inventoried. Unselected zones are transitioned as opposed to inventoried. (c) Inventory progress view. It is divided in two blocks. Mission shows global information. It lists the zones configured in the current mission (C2-C3), total duration and status. The second block, Current area, shows information about the ongoing inventory, which corresponds to a single zone. 51

4.14 Interface views under the Reports menu. (a) Reports is divided in two submenus: Tasks history and Maps information. (b) Tasks history allows the consultation of every task undertaken by the robot. (c) Maps information lists configured zones and whether the recognition has been completed (Map date). As well, it suggests rerunning the recognition when significant layout changes are detected (Remapping). 52

4.15 (a) Robot's control interface Administration menu. (b) Teleoperation allows the remote commanding of the robot and shows a live stream of the camera as reference. (c) Diagnostics view is intended for troubleshooting. It lists the status of all relevant hardware parts. 54

5.1 Data subsets considered at accuracy computation. Four zones are depicted, expressed by their corresponding product types: Z_1, Z_2, Z_3 and Z_4. The inventoried zone in this case is Z_2. The set of handheld detections is H_2, the set of robot detections is R_2 and the perpetual inventory is P_2. Usually, both the robot and the handheld identify items that belong to zones adjacent to the zone in scope. In the case of the robot, this is usually more significant. 60

5.2 Visualisation of the different baselines - delimited with a dashed red line - used in accuracy computation. The use of one or other baseline measure depends on data availability. (a) Raw baseline (B_{raw}). (b) Filtered baseline ($B_{filtered}$). (c) Verified baseline ($B_{verified}$). 64

5.3 Within the same area, layout (b) includes more aisles length than layout (a). Therefore, the degree of intricacy in layout (b) is higher than in layout (a). Note that a complete navigation of layout (b) involves three turns in narrower aisles in contrast with the two turns and wider aisles in layout (a). 66

5.4 The evolution of accuracy over time in a real-world scenario. A fast growth is followed by a slow growth. Around time 2000 s the discovery of a new crowded area restarts the typical growth sequence. Effort does not contribute linearly to accuracy. Note the two quasi-flat regions, where a prolonged time period is needed to gain a small share of accuracy. The explanation is that easy identifications happen in bulks while difficult ones need a continued effort. 69

5.5 The robot at the verification setting, the University Library. Side covers removed so six RFID antennas are visible on one side. 72

5.6 SGTIN-96 coding scheme, the one used to code books in the library setting. Source: https://www.gs1.org/ 74

5.7 RFID-Driven navigation control verification tests results. Three different configurations were tested, which are summarised in Table 5.3. Each configuration involves four aisle passes. Each pass contributes to the accuracy in a cumulative way, new identifications are added to former passes. Seven tests were completed under each configuration and the mean (symbol) and span (lines) of the accuracy are assessed. The dotted horizontal line at 99% marks the minimum acceptable accuracy. 76

5.8 RFID-Driven navigation control optimisation results. The optimisation focus is in the comprise between inventory accuracy and duration. Each pair of thresholds involves four aisle passes, marked by four consecutive symbols. Each pass contributes to the accuracy in a cumulative way, new identifications are added to former passes. Seven tests were completed under each configuration and the mean (symbol) and span (lines) of the accuracy are assessed. The dotted horizontal line at 99% marks the minimum acceptable accuracy. The dotted vertical line at 600 s marks the baseline duration, the best measured during verification tests, $(th_{twist} = 1, th_{journey} = 0)$. 79

5.9 Summary of validation measures. On the first row, zone characteristics: (a) density of items, (b) aisles length and (c) intricacy. On the second row, inventory measures: (d) inventory duration, (e) inventory distance and (f) inventory speed. On the third row, figures of merit: (g) filtered accuracy, (h) effective speed and (i) read speed. Plots are actually a graphical display of tabular data. 88

6.1 University Library layout where the RFID location dataset was gathered. 91

6.2 Representative binned identification streams that support the similarity coefficients discussion. The identifications' count in each bin is depicted for four items. 104

6.3 Dendogram of a subset of Violent Crime Rates by US State dataset[1]. The vertical axis shows the distance between samples grouped by the dendogram branches. The red dashed line marks the height at which the dendogram is cut to form 4 clusters, shown in different colours. . . . 106

6.4 Schematics of an aisle perspective composed by two shelf racks divided in book blocks. Highlighted, (a) Subset A blocks and (b) Subset B blocks. Subset B is more challenging than Subset A for clustering because of the proximity of blocks. 110
6.5 Results of clustering subset A book blocks (classes). . . 112
6.6 Dendogram of Subset A under the best configuration, which is achieved by all the similarity coefficients, many bin widths and method ward.D2. The colours on the dendogram depict the 5 resulting clusters after cutting the tree. Below, the samples assigned to each cluster are coloured and named after their actual book block, their actual class. The dendogram shows a perfect clustering for Subset A. 113
6.7 Results of clustering subset B book blocks (classes). . . . 114
6.8 Dendogram of Subset B under the best configuration: similarity coefficient WWSMC; method ward.D2; and bin width 1 second. The colours on the dendogram depict the 5 resulting clusters after cutting the tree. Below, the samples assigned to each cluster are coloured and named after their actual book block, their actual class. There are 4 samples that do not coincide with the majority within the class (colour mismatch). Anyway, the dendogram shows a good clustering, 70 out of 74 samples are assigned to the correct class, for Subset B. 115
6.9 Results of clustering Subset 50 book blocks (classes). . . 116
6.10 Results of clustering Subset 50 book blocks (classes) considering samples assigned to a block contiguous, along the shelf, to their actual block as correctly classified. . . 117
6.11 Results of clustering Subset 50 book blocks (classes) considering samples assigned to a block contiguous, along the shelf, to their actual block as correctly classified. . . 118
6.12 Results of clustering subset 50 book blocks (classes) considering samples assigned to a block contiguous, along or across the shelf, to their actual block as correctly classified. 119

6.13 Results of clustering all the books in the library, considering samples assigned to a block contiguous, along or across the shelf, to their actual block as correctly classified. 121

6.14 Cullen and Frey graph of the samples of inventory round inv_5. The plot shows that the probability distribution of the identifications is not close to a Gaussian. 126

6.15 Cullen and Frey graph of the samples of inventory round inv_5 for each relative antenna-item location. The plot show the density of points falling on each region. There is not a distribution that characterises univocally the different antenna-item locations. 127

6.16 Identification models computed with the combination of $K = \{1, 5, 10\}$ and $D_{max} = \{60, 120, 180\}$ for a single antenna and height. The black crosses mark identifications, which can overlap. The coloured dots represent the $rssi$ mean of the computed identification model. The models depicted grid resolution is 30 cm. A 2-D slice (one height) of the complete 3-D model is shown for visualization purposes. 130

6.17 Histograms of the 3-D location euclidean error for the different combinations of model parameters considering a Uniform RSSI probability distribution. The grid is composed of: horizontally $D_{max} = (60, 120, 180)$; and vertically $N_{max} = (1, 5, 10)$ samples. The red dashed line marks the error mean and the blue line depicts the cumulative histogram. The lowest error is given by the combinations ($D_{max} = 120$, $N_{min} = 5$) and ($D_{max} = 180$, $N_{min} = 10$). 133

6.18 Histograms of the 3-D location euclidean error for the different combinations of model parameters considering a Gaussian RSSI probability distribution. The grid is composed of: horizontally $D_{max} = (60, 120, 180)$; and vertically $N_{max} = (1, 5, 10)$ samples. The red dashed line marks the error mean and the blue line depicts the cumulative histogram. The lowest error is given by the combinations ($D_{max} = 120$, $N_{min} = 5$) and ($D_{max} = 180$, $N_{min} = 10$). 134

6.19 Relation between the euclidean error of the location and the number of identification instances for each object detected. One point of the plot represents one object. In blue, the loess regression of the samples. There is a minimum around 10 instances per object. The plot shown corresponds to model parameters ($D_{max} = 180$, $N_{min} = 10$) and a uniform RSSI distribution. 136

6.20 Relation between the aggregated standard deviation of robot positions that originate detection instances and number of instances for each object detected. One point of the plot represents one object. The higher the number of instances, the more diversity in robot positions. The plot shown corresponds to model parameters ($D_{max} = 180$, $N_{min} = 10$) and a uniform RSSI distribution. 137

A.1 Left bottom antenna identification streams of books in Subset A. Each line represents the identification instances of a single book. Blocks are grouped by colour. 146

A.2 Right middle antenna identification streams of books in Subset A. Each line represents the identification instances of a single book. Blocks are grouped by colour. 147

A.3 Left bottom antenna identification streams of books in Subset B. Each line represents the identification instances of a single book. Blocks are grouped by colour. 148

A.4 Right middle antenna identification streams of books in Subset B. Each line represents the identification instances of a single book. Blocks are grouped by colour. 149

B.1 View of a 3D occupancy map recorded by the robot. Colour is used as a height scale. Blue is ground level. 151

List of Tables

3.1 Results of inventorying with the preliminary humanoid robot design. 21

4.1 Contribution of subsystems to requirements 27
4.2 Results of the questionnaire for the validation of robot assembly. The statements were valued between 1 (totally disagree) and 5 (totally agree). 55
4.3 Results of the questionnaire for the validation of robot operation. The statements were valued between 1 (totally disagree) and 5 (totally agree). 55

5.1 Summary of accuracy measures and the data sets required to compute them. 61
5.2 Characteristics of the verification setting. 72
5.3 Configurations of the RFID-driven navigation control verification tests. 75
5.4 RFID-Driven navigation control verification tests results. Three different configurations are shown, which are summarised in Table 5.3. Each pass contributes to the accuracy in a cumulative way, new identifications are added to former passes. Accuracy Acc refers to the verified accuracy. 77
5.5 Robot (Configuration C, 2 passes) and handheld (4 passes) figures of accuracy and duration measured during the verification tests. Accuracy Acc refers to the verified accuracy. 78

5.6 Figures of merit of the optimal robot configuration and the handheld in the library. The optimal robot configuration is the one with $(th_{twist} = 16, th_{journey} = 8)$ and two passes. 80
5.7 Main robot field experimentation periods at retailers. . . 83
5.8 Characteristics of the store sections where experimental tests were conducted. 84

6.1 The location dataset consists of five inventory rounds under different configurations. Configurations using session S1 yield a higher number of identifications per book than configurations using session S2. This implies a higher diversity of robot positions, which is assumed to contribute positively to location algorithms. 93
6.2 List of data included in the location dataset 95
6.3 Similarity coefficients corresponding to the four items whose identification streams are depicted in Figure 6.2. 103
6.4 Agglomerative hierarchical clustering methods. A and B represent two clusters, $D(A, B)$ the distance between them, and $d(a, b)$ the distance between two members of a cluster. 107
6.5 Parameters and values that are combined for clustering evaluation. 108
6.6 Example of clustering two classes, A and B, and the corresponding confusion matrix. 109
6.7 Summary of location results for the different combination of parameters analysed. For simplicity, it includes the most significant combinations of parameters, the ones that offer a better performance in their group of analysis. Reducing the number of detection instances in the computation of locations, reduces significantly the error. The minimum error is given by using the 10 identifications instances with the highest RSSI. 135

References

[1] D. J. Bowersox, D. J. Closs, M. B. Cooper, and J. C. Bowersox, *Supply chain logistics management*. McGraw-Hill, New York, 4th ed., 2013.

[2] C. Dubelaar, G. Chow, and P. D. Larson, "Relationships between inventory, sales and service in a retail chain store operation," *International Journal of Physical Distribution & Logistics Management*, vol. 31, no. 2, pp. 96–108, 2001.

[3] A. Raman, N. DeHoratius, and Z. Ton, "Execution: The missing link in retail operations," *California Management Review*, vol. 43, no. 3, pp. 136–152, 2001.

[4] Y. Kang and S. B. Gershwin, "Information inaccuracy in inventory systems: stock loss and stockout," *IIE Transactions*, vol. 37, no. 9, pp. 843–859, 2005.

[5] A. G. Kök and K. H. Shang, "Inspection and replenishment policies for systems with inventory record inaccuracy," *Manufacturing & Service Operations Management*, vol. 9, no. 2, pp. 185–205, 2007.

[6] D. A. Schrady, "Operational definitions of inventory record accuracy," *Naval Research Logistics Quarterly*, vol. 17, no. 1, pp. 133–142, 1970.

[7] E. Sahin and Y. Dallery, "Assessing the impact of inventory inaccuracies within a newsvendor framework," *European Journal of Operational Research*, vol. 197, no. 3, pp. 1108 – 1118, 2009.

[8] S. Cannella, J. M. Framinan, M. Bruccoleri, A. P. Barbosa-Póvoa, and S. Relvas, "The effect of inventory record inaccuracy in information exchange supply chains," *European Journal of Operational Research*, vol. 243, no. 1, pp. 120 – 129, 2015.

[9] Y. Ettouzani, N. Yates, and C. Mena, "Examining retail on shelf availability: promotional impact and a call for research," *International Journal of Physical Distribution & Logistics Management*, vol. 42, no. 3, pp. 213–243, 2012.

[10] E. Fleisch and C. Tellkamp, "Inventory inaccuracy and supply chain performance: a simulation study of a retail supply chain," *International Journal of Production Economics*, vol. 95, no. 3, pp. 373 – 385, 2005.

[11] M. Bruccoleri, S. Cannella, and G. L. Porta, "Inventory record inaccuracy in supply chains: the role of workers' behavior," *International Journal of Physical Distribution & Logistics Management*, vol. 44, no. 10, pp. 796–819, 2014.

[12] H. H.-C. Chuang and R. Oliva, "Inventory record inaccuracy: Causes and labor effects," *Journal of Operations Management*, vol. 39-40, pp. 63 – 78, 2015.

[13] T. J. Kull, M. Barratt, A. C. Sodero, and E. Rabinovich, "Investigating the effects of daily inventory record inaccuracy in multichannel retailing," *Journal of Business Logistics*, vol. 34, no. 3, pp. 189–208, 2013.

[14] S. Cannella, R. Dominguez, and J. M. Framinan, "Inventory record inaccuracy - The impact of structural complexity and lead time variability," *Omega*, vol. 68, pp. 123 – 138, 2017.

[15] H. Sandoh and H. Shimamoto, "A theoretical study on optimal inventory-taking frequency for retailing," *Journal of Retailing and Consumer Services*, vol. 8, no. 1, pp. 47–52, 2001.

[16] P. S. Neeley, "A framework for cycle counting," *Production and Inventory Management*, vol. 24, no. 4, pp. 23–32, 1983.

[17] S. Perrey, T. Thedon, and T. Rupp, "NIRS in ergonomics: Its application in industry for promotion of health and human performance at work," *International Journal of Industrial Ergonomics*, vol. 40, no. 2, pp. 185 – 189, 2010.

[18] H. J. Einhorn and R. M. Hogarth, "Confidence in judgment: Persistence of the illusion of validity.," *Psychological Review*, vol. 85, no. 5, pp. 395–416, 1978.

[19] R. Want, "An introduction to RFID technology," *IEEE Pervasive Computing*, vol. 5, pp. 25–33, January 2006.

[20] ThingMagic, a JADAK Brand , "The Mercury6e (M6e) embedded UHF RFID reader module." http://www.thingmagic.com/index.php/embedded-rfid-readers/mercury6e, 2017. Last accessed on March 03, 2017.

[21] GS1, "EPC UHF Gen2 Air Interface Protocol." https://www.gs1.org/epcrfid/epc-rfid-uhf-air-interface-protocol/2-0-1, 2017. Last accessed on September 18, 2018.

[22] B. C. Hardgrave and J. Patton, "State of RFID adoption among U.S. apparel retailers." https://rfid.auburn.edu/papers/2016-state-rfid-adoption-among-u-s-apparel-retailers/, 2016. Last accessed on September 18, 2018.

[23] GS1 US, "GS1 US survey shows manufacturers and retailers embrace RFID to enhance inventory visibility."

https://www.gs1us.org/what-we-do/about-gs1-us/media-center/press-releases/detail/articleid/60/gs1-us-survey-shows-manufacturers-and-retailers-embrace-rfid-to-enhance-inventory-visibility, 2015. Last accessed on September 19, 2018.

[24] Impinj, "Impinj Makes Milestone Shipping 25 Billionth RAIN RFID Endpoint IC." https://www.impinj.com/library/blog/impinj-makes-milestone-shipping-25-billionth-rain-rfid-endpoint-ic, 2018. Last accessed on September 18, 2018.

[25] PAL Robotics, "Stockbot: Automatic inventory." http://pal-robotics.com/en/products/stockbot. Last accessed on September 18, 2018.

[26] MetraLabs, "Tory. Efficient automated RFID inventory." https://www.metralabs.com/en/rfid-robot-tory/, 2017. Last accessed on September 18, 2018.

[27] Simbe Robotics, "Meet tally. The world's first fully autonomous shelf auditing and analytics solution." http://www.simberobotics.com/. Last accessed on September 18, 2018.

[28] I. Ehrenberg, C. Floerkemeier, and S. Sarma, "Inventory management with an RFID-equipped mobile robot," in *2007 IEEE International Conference on Automation Science and Engineering*, pp. 1020–1026, 2007.

[29] T. Schairer, C. Weiss, P. Vorst, J. Sommer, C. Hoene, W. Rosenstiel, W. Strasser, A. Zell, G. Carle, P. Schneider, and A. Weisbecker, "Integrated scenario for machine-aided inventory using ambient sensors," in *4th European Workshop on RFID Systems and Technologies*, pp. 1–8, VDE Verlag, June 2008.

[30] K. Nur, M. Morenza-Cinos, A. Carreras, and R. Pous, "Projection of RFID-obtained product information on a retail stores indoor panoramas," *IEEE Intelligent Systems*, vol. 30, pp. 30–37, November 2015.

[31] J. Zhang, Y. Lyu, T. Roppel, J. Patton, and C. P. Senthilkumar, "Mobile robot for retail inventory using RFID," in *2016 IEEE International Conference on Industrial Technology (ICIT)*, pp. 101–106, IEEE, March 2016.

[32] M. Morenza-Cinos, V. Casamayor-Pujol, J. Soler-Busquets, J. L. Sanz, R. Guzmán, and R. Pous, "Development of an RFID inventory robot (AdvanRobot)," in *Robot Operating System (ROS). The Complete Reference (Volume 2)* (A. Koubaa, ed.), pp. 387–471, Berlin: Springer, 2017.

[33] M. Bertolini, E. Bottani, G. Romagnoli, and G. Vignali, "The impact of RFID technologies on inventory accuracy in the apparel retailing: Evidence from the field," *International Journal of RF Technologies*, vol. 6, no. 4, pp. 225–246, 2015.

[34] A. Rizzi and G. Romagnoli, *Testing and Deploying an RFID-Based Real-Time Locating System at a Fashion Retailer: A Case Study*, pp. 201–214. Cham: Springer, 2017.

[35] R. Miesen, R. Ebelt, F. Kirsch, T. Schäfer, G. Li, H. Wang, and M. Vossiek, "Where is the tag?," *IEEE Microwave Magazine*, vol. 12, pp. S49–S63, December 2011.

[36] D. Joho, C. Plagemann, and W. Burgard, "Modeling RFID signal strength and tag detection for localization and mapping," in *2009 IEEE International Conference on Robotics and Automation*, pp. 3160–3165, May 2009.

[37] R. Liu, A. Koch, and A. Zell, "Mapping UHF RFID tags with a mobile robot using a 3D sensor model," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1589–1594, November 2013.

[38] A. Koch and A. Zell, "Mapping of passive UHF RFID tags with a mobile robot using outlier detection and negative information," in

2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 1619–1624, IEEE, May 2014.

[39] R. Miesen, F. Kirsch, and M. Vossiek, "UHF RFID localization based on synthetic apertures," *IEEE Transactions on Automation Science and Engineering*, vol. 10, pp. 807–815, July 2013.

[40] H. Ma, Y. Wang, K. Wang, and Z. Ma, "The optimization for hyperbolic positioning of UHF passive RFID tags," *IEEE Transactions on Automation Science and Engineering*, vol. 14, pp. 1590–1600, October 2017.

[41] T. Liu, L. Yang, Q. Lin, Y. Guo, and Y. Liu, "Anchor-free backscatter positioning for RFID tags with high accuracy," in *IEEE Conference on Computer Communications (INFOCOM)*, pp. 379–387, IEEE, April 2014.

[42] M. Cremer, U. Dettmar, C. Hudasch, R. Kronberger, R. Lerche, and A. Pervez, "Localization of passive UHF RFID tags using the AoAct transmitter beamforming technique," *IEEE Sensors Journal*, vol. 16, pp. 1762–1771, March 2016.

[43] M. Scherhäufl, M. Pichler, and A. Stelzer, "Maximum likelihood position estimation of passive UHF RFID tags based on evaluation of backscattered transponder signals," in *2016 IEEE Topical Conference on Wireless Sensors and Sensor Networks (WiSNet)*, pp. 24–26, IEEE, January 2016.

[44] D. Hahnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose, "Mapping and localization with RFID technology," in *2004 IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, pp. 1015–1020, IEEE, April 2004.

[45] P. Vorst, S. Schneegans, B. Yang, and A. Zell, "Self-localization with RFID snapshots in densely tagged environments," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1353–1358, IEEE, September 2008.

[46] H. Chae and K. Han, "Combination of RFID and vision for mobile robot localization," in *2005 International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 75–80, IEEE, December 2005.

[47] T. Kämpke, B. Kluge, and M. Strobel, "Exploiting RFID capabilities onboard a service robot platform," in *Towards Service Robots for Everyday Environments: Recent Advances in Designing Service Robots for Complex Tasks in Everyday Environments* (E. Prassler, M. Zöllner, R. Bischoff, W. Burgard, R. Haschke, M. Hägele, G. Lawitzky, B. Nebel, P. Plöger, and U. Reiser, eds.), (Berlin, Heidelberg), pp. 215–225, Springer, 2012.

[48] T. Deyle, M. S. Reynolds, and C. C. Kemp, "Finding and navigating to household objects with UHF RFID tags by optimizing RF signal strength," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2579–2586, IEEE, September 2014.

[49] A. Kleiner, J. Prediger, and B. Nebel, "RFID technology-based exploration and SLAM for search and rescue," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4054–4059, IEEE, October 2006.

[50] M. Morenza-Cinos, "RFID location dataset." http://dx.doi.org/10.4225/13/511C71F8612C3.

[51] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2, pp. 142–149, IEEE, 2000.

[52] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, USA: MIT Press, 2012.

[53] L. L. McQuitty, "Single and multiple hierarchical classification by reciprocal pairs and rank order types," *Educational and Psychological Measurement*, vol. 26, no. 2, pp. 253–265, 1966.

[54] Z. Chen *et al.*, "Bayesian filtering: From Kalman filters to particle filters, and beyond," *Statistics*, vol. 182, no. 1, pp. 1–69, 2003.

[55] P. Vorst and A. Zell, *Semi-autonomous learning of an RFID sensor model for mobile robot self-localization*, pp. 273–282. Berlin, Heidelberg: Springer, 2008.