UNIVERSITÀ DI PARMA

UNIVERSITÀ DEGLI STUDI DI PARMA

Dottorato di Ricerca in Tecnologie dell'Informazione

XXXII Ciclo

Modern Automation Challenges:
Towards Seamless AGV Localization

Coordinatore:

Chiar.mo Prof. Marco Locatelli

Tutor:

Chiar.mo Dott. Dario Lodi Rizzini

Dottorando: *Francesco Galasso*

Anni 2016 / 2019

"É sempre l’altra…"

Abstract

Modern warehouses constantly raise in complexity, widely spreading across many industrial fields. High flow efficiency, operation assurance, significant flexibility and pin-point repeatability are compelling in automated factories.

Autonomous Guided Vehicles (AGVs) fleets are replacing manual lift trucks, guaranteeing solid warehousing performance and safe behaviours towards people, structures and other moving objects. They are usually equipped with forks to move products rapidly and safely.

To correctly handle industrial materials, these vehicles require an estimation of position and orientation with accuracy respectively below 2 *cm* and 1 *deg*, especially during pick-up and deployment actions. Furthermore, AGVs ought to incorporate versatile localization strategies to be able to navigate in many different industrial contexts. For this reason, each vehicle is equipped with a localization laser rangefinder to accurately perceive the surrounding environment. Inaccurate AGVs positioning, especially between vehicles of the same fleet, or uncertain localization performance may lead to stock damages or people injuries.

The first contribution of this dissertation is the proposal of a novel automatic calibration method for four wheel AGVs. Parameter tuning is essential to guarantee that each vehicle of the fleet can reach the same point of the warehouse with adequate precision. Usually, the calibration of AGVs is an iterative manual activity: the operator runs a series of tests and tunes one parameter at a time. This operation requires a large amount of time, can be inaccurate, and strongly depends on the experience of the technician. The proposed calibration method estimates both the kinematic parameters and the laser rangefinder parameters by comparing the sensor egomotion and the motion commands. This calibration procedure is considerably faster, completely automatic, and independent for the ability of the operator, being able to estimate all the parameters at the same time with higher accuracy. This method has been tested on field on many different AGVs of different fleets to assess its correctness and to evaluate its actual performance.

The second contribution of this thesis is the development of three different methods for localization in unstructured environments. Usually, the standard AGV localization is performed using the laser rangefinder to detect artificial landmarks, matching them with a static reflector map. The installation of those landmarks is expensive and time-consuming, as well as inflexible w.r.t. environmental changes. The reconstruction of industrial environments is built using a Terrestrial Laser Scanner (TLS) as part of an advanced plant setup workflow. Thereafter, a 2D map can be extracted from the TLS survey with advanced software processing. The localization methods proposed in this dissertation exploit such map either to extract

environmental landmarks or to perform dense map registration. With these methods, no artificial landmark has to be deployed, so system installation becomes faster and less expensive. These localization approaches for unstructured environments return better or similar performances w.r.t. the canonical artificial landmark approach. Such methods have been tested on real AGVs in an industrial warehouse to assess their performance.

The third contribution of this dissertation is the proposal of a novel signature for place recognition and loop closure with landmark maps, named GRD (*Geometric Relation Distribution*). Precise location signature is essential to reduce localization errors and inconsistencies during SLAM (Simultaneous Localization And Mapping) procedures and can also be used for Global Localization during AGV pose initialization. The proposed signature is suitable for artificial landmarks maps, like those currently used in industrial localization systems, as well as for feature-based maps. Performance have been evaluated through experiments on standard datasets.

The combination of advanced automatic calibration methods, flexible localization strategies and place recognition approaches developed in this thesis can remarkably increase on-point localization performance, greatly reduce plant installation times, and vastly enhance warehouse efficiency.

Contents

Chapter 1

Introduction

This chapter describes the main contributions of this thesis, both theoretical and practical. The industrial issues that motivated this work are presented in Section 1.1. The contributions and the novelty of the proposed solutions are discussed in Section 1.2. Finally, a brief overview of the following chapters is provided in Section 1.3.

1.1 Motivation

This thesis focuses on improving positioning performance of Autonomous Guided Vehicles (AGVs). High accuracy and pin-point precision are key features for vehicle positioning to ensure efficient warehouse management, avoiding product damages and dangerous accidents. Furthermore, different types of AGVs have to co-exist in the same industrial plant and they can be equipped alternatively with forks, hooks, clamps, conveyors or pantographs. Even if the material movement device is different, those AGVs must co-operate properly in order to guarantee the most regular and efficient flow of goods.

An important aspect to satisfy these requirements is AGV calibration. To correctly move materials, those vehicles require position and orientation accuracy respectively below 2 *cm* and 1 *deg*. Typically, an operator performs a manual iterative procedure tuning AGV parameters, one at a time. Both the kinematic parameters of the vehicle and the pose of the localization laser rangefinder have to be tuned. This operation is time-consuming and error-prone, being very dependent on the ability and the experience of the technician. Errors during manual calibration are quite frequent and can lead to vehicle accidents, product damages and people injuries, while also decreasing plant efficiency. Furthermore, AGV calibration is currently performed during the installation of the automated warehouse, which is the most critical phase of the project; any delay in this stage may unpleasantly impact on customer production activities.

Another important element regarding AGV positioning is the localization method. The standard and most common solution for AGV localization exploits a dedicated laser rangefinder to detect artificial landmarks, named *reflectors*. Subsequently, these landmarks are matched with a static map. Other SLAM-based localization approaches are not common since the mapping phase is crucial to obtain adequate positioning performance. Because of the high complexity of modern warehouses a large number of reflectors must be installed, increasing deployment times and costs. In addition, such undesirable effects scale badly with plant size. Furthermore, this solution is not flexible w.r.t. environmental changes, since artificial landmarks must be carefully mapped every time they are added or moved. Likewise AGV calibration, the installation and mapping of reflectors is performed during the installation of the automated warehouse, when any delay should be avoided.

The investigation reported in this thesis has been supported by Elettric80 SpA, a company founded in the 1980s in Viano, close to Reggio Emilia, Italy. The core business of the company lies in tailor-made solutions to manage supply chain activities, improving factory efficiency and enhancing product traceability. The main systems include AGVs, palletizing robots, wrappers, labellers, repacking systems, high density warehouses and other end-of-line solutions. With these technologies, flexible, modular and integrated solutions can be developed for high-volume manufacturers within different sectors like beverage, food, tissue and ceramics. A centralized software platform manages the entire logistic flow, optimizing all operations from raw materials reception to product shipping.

As of today, Elettric80 SpA has completed approximately 300 automated warehouses all around the world and installed over 2000 robotics systems coupled with over 4500 AGVs. In addition to the central headquarter in Viano, the company has 13 branches worldwide, gathering around 720 employees.

Figure 1.1: Different Autonomous Guided Vehicles (AGVs) manufactured by Elettric80 SpA with different material movement tools installed (forks, pantographs, conveyors or clamps).

1.2 Contribution

The main contributions of this dissertation are the following:

- A novel automatic calibration method for two different four wheel kinematics of AGV is presented. This procedure tunes simultaneously both the kinematic parameters of the AGV (named *intrinsic parameters*) and the laser scanner parameters (named *extrinsic parameters*). The calibration procedure compares the sensor egomotion and the odometric measures to estimate the desired parameters. Each vehicle configuration is modelled, discussed and solved relatively to the tuning parameters. The benefits w.r.t. the traditional manual calibration procedure are also shown. Moreover, a vast set of experiments on single AGV and on AGV fleets is presented and discussed.

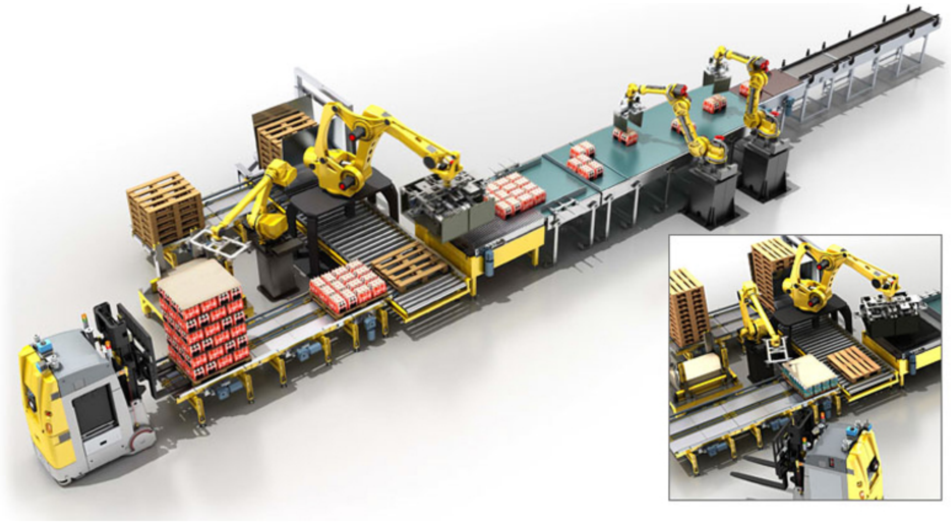


Figure 1.2: Palletizing cell manufactured by Elettric80 SpA, featuring robot arms to form layers of product and stack layer over layer on a pallet, picked up by an AGV with forks.

- Three novel localization methods for AGVs in unstructured environments are presented. These techniques allow vehicle localization without deploying any artificial landmark in the industrial plant. A precise 2D map of the environment is obtained using a Terrestrial Laser Scanner (TLS) and advanced algorithms. The first method detects environmental keypoints that are matched with high curvature regions of the map of the plant. The latter two methods estimate the pose of the AGV comparing a laser scan and the precomputed dense map. The advantages w.r.t. the well-established reflector localization are described. Many localization experiments are discussed to assess the accuracy of the proposed methods.

- All the aforementioned solutions to improve AGV positioning have been implemented and tested on real AGVs, to fully evaluate performance improvements. Differently from many other approaches, those methods embed high technology readiness levels and match industrial safety requirements, while also being highly innovative in the industrial field.

- A novel signature for place recognition and loop closure in landmark maps named GRD (*Geometric Relation Distribution*) is presented. This approach can be used with both reflectors and environmental keypoints to evaluate SLAM-like techniques. The effectiveness of this strategy has been determined using standard datasets.

1.3 Thesis Outline

The thesis is organized as follows: in Chapter 2, a typical AGV warehouse is described, focusing on the kinematic characteristics of the vehicles and the sensing technologies they embed. Issues and limitations of the current solutions are also discussed w.r.t. the proposed methods. Chapter 3 presents the novel automatic calibration method for four wheel vehicles. In Chapter 4, three novel localization methods for industrial AGVs are presented. Chapter 5 presents the novel GRD signature for place recognition and loop closure. Finally, conclusion and future work are discussed in Chapter 6.

Chapter 2

Technologies for Industrial Logistics

This chapter presents a brief overview of industrial automated systems that exploit a fleet of AGVs, deepening its modelling characteristics and sensing technologies. Critical issues, technological limitations and novel solutions are also discussed.

A brief survey of the state of the art in industrial logistics is reported in Section 2.1. A typical automated warehouse handled by AGVs and its characteristics are presented in Section 2.2. The main aspects of AGV modelling and the most used kinematics are discussed in Section 2.3. The most common AGV sensing technologies, mapping techniques and localization approaches are described in Section 2.4. Finally, a possible workflow for automated warehouse design exploiting the contributions of this thesis is presented in 2.5.

2.1 AGVs in Industrial Logistics: a Brief Survey

In modern industrial plants, automation and robotics are widely spreading among many production phases, including raw materials management, end of the line processing and semi-finished product handling. The same trend can be observed in logistics, where modern automated solutions like AGV fleets are replacing human operators and manual vehicles. Indeed, the deployment of AGVs aims to increase the productivity of the plant and the efficiency of the warehouse. Safety and repeatability are also boosted while down-times and product damages are drastically reduced.

An industrial plant exploiting AGVs and its core characteristics is described in [1,2]. Their sensing technologies, navigation strategies and material handling capabilities are presented and the main differences with human-driven vehicles are highlighted. Sabattini *et al.* [3] describe the currently applied solutions in the AGV field and a possible roadmap to improve system performance. These modern approaches include traffic management enhancements, more advanced sensing systems and improved navigation approaches. In [4] an overview of an industrial plant with AGVs is presented, focusing on advanced sensing technologies and efficient coordination techniques. The application of modern methods leads to improved performance and higher awareness of the vehicles. D'Andrea *et al.* [5] further investigate the coordination problem focusing on wireless communication, advanced sensors, efficient resource allocation and careful status monitoring. The largest installation to date powered by the aforementioned approach embeds more than 500 robots. In [6] a practical solution to embed an AGV fleet into a pre-existent plant is discussed and tested, focusing on localization, mapping, safety and deployability. An optimized workflow to integrate mobile robots in retail stores and supermarkets is presented in [7]: goals such as human-robot collaboration, efficiency boosting and fast refilling are reached exploiting stocks monitoring, goods pre-sorting and semi-automatic shelf filling.

Despite the technological level reached in the last years, much room for improvement is still available: localization, coordination, flexibility and safety can progress. In fact, this thesis will focus on some of the aforesaid topics.

2.2 Case Study: a Navigation System for AGVs

As previously stated, AGVs are most often used in industrial applications to move materials around vast and complex factories or warehouses. The total number of vehicles and the choice of the appropriate type of AGV depend on the size of the warehouse and the amount of goods flowing within the industrial plant. Hence after, a case study of AGV navigation system inspired by the logistic plants designed by Elettric80 SpA is presented. Such case is sufficiently general to serve as conceptual model for automated warehouses.

A centralized Traffic Manager (*TM*) handles material flows and efficiently generates the missions for each AGV in the fleet. When missions are assigned, AGVs follow pre-calculated paths called *segments* to reach the *operation points*, where materials are deployed or picked-up. The implementation with centralized Traffic Manager and pre-computed *segments* is the most common in industrial scenarios assuring high performance, highly safe movements and rapid deployment. Figures 2.1 and 2.2 illustrate two examples of industrial warehouse layouts.

While driving on *segments* or halting at *operation points*, AGVs must know very accurately their position. This is crucial since vehicles have to drive fast enough to ensure high plant efficiency and, at the same time, they must be safe w.r.t. objects, people and other vehicles. Furthermore, different AGVs within the same fleet (even with different kinematics or other lifting tools than forks) must operate together to handle materials fast, efficiently and with high repeatability. Precise positioning is then one of the most essential feature for an AGV fleet.

Every AGV manufacturer wishes to develop advanced methodologies to increase the global efficiency of the system, leading to higher productive rates. This can be obtained in many different ways, such as:

- Automating the operations as much as possible to become independent from operators and their individual skill level;
- Uniforming the positioning behaviour of the AGVs belonging to the fleet;
- Enhancing the flexibility of the fleet in different industrial scenarios;
- Reducing the deployment time of the automated warehouse and the impact on customer operations.

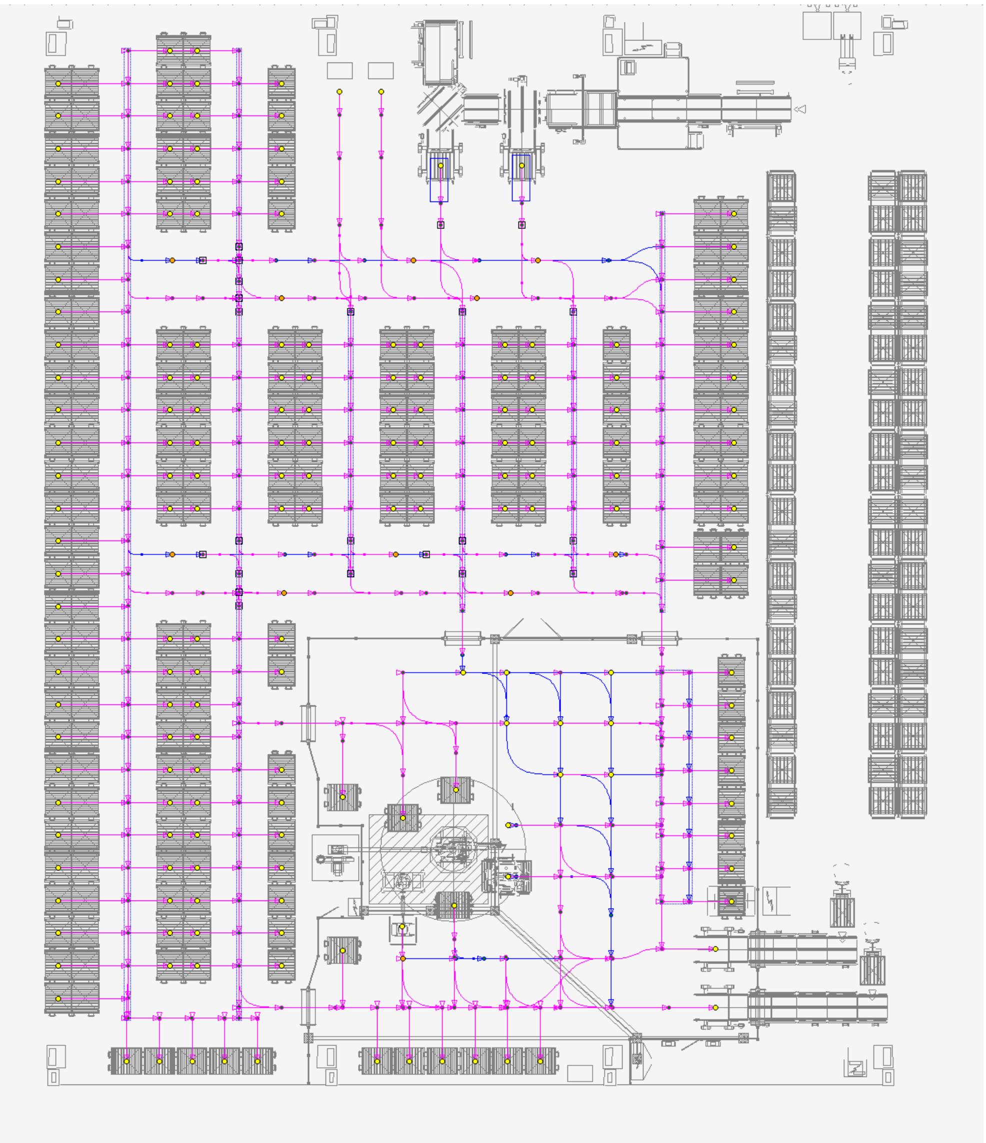


Figure 2.1: A simple AGV layout of an industrial plant with designed *segments* (blue and pink) and *operation points* (yellow).

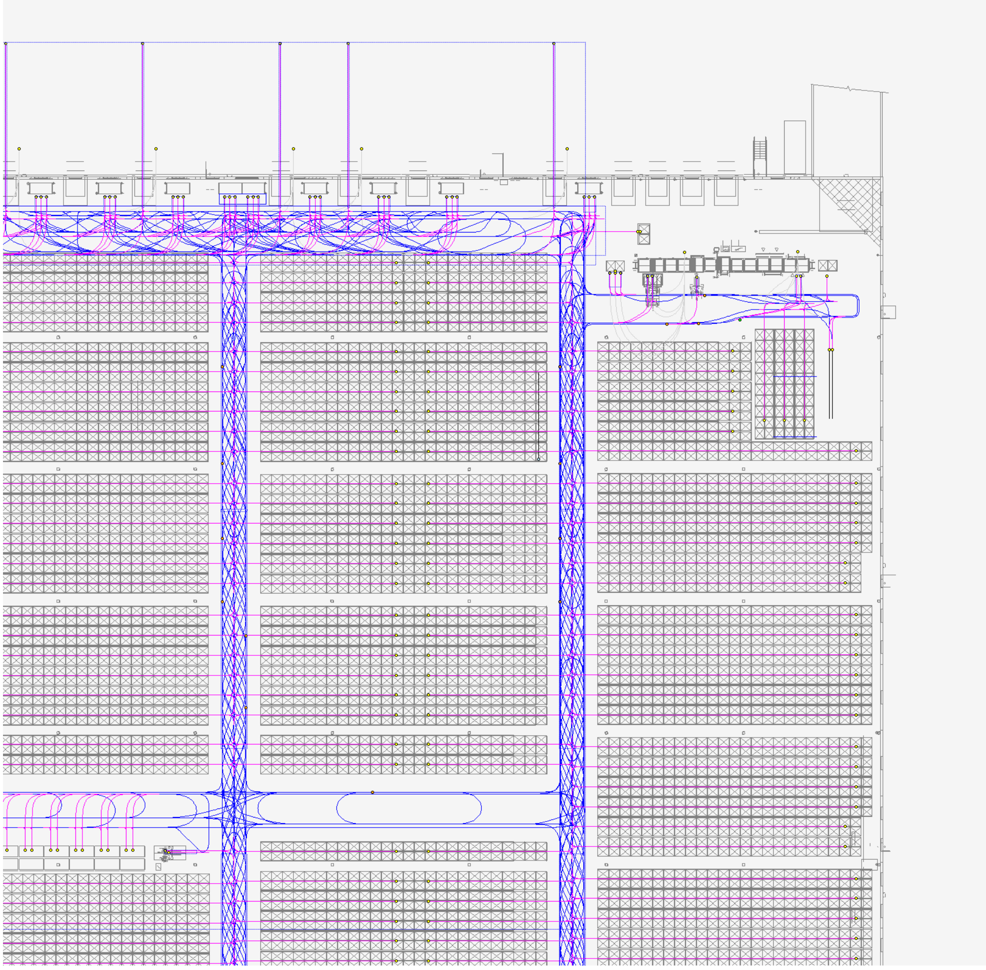
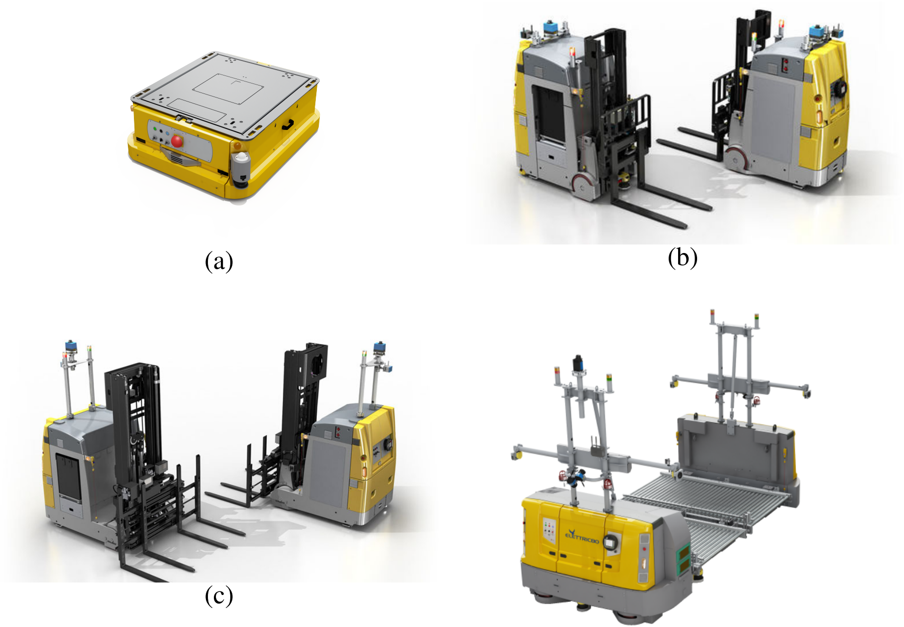


Figure 2.2: Detail of an AGV layout of an industrial plant with designed *segments* (blue and pink) and *operation points* (yellow). Note the density and the complexity of the paths.

2.3 Classification and Modelling of AGVs

Many different types of AGV are manufactured to fulfil different tasks. Vehicles can mount different lifting solutions like forks, clamps, hooks, pantographs or conveyors.

AGVs can also have different kinematics like differential drive, three and four wheel vehicles, both holonomic and nonholonomic. Some different AGV kinematics are illustrated in Figure 2.3.



(a) (b) (c) (d)

Figure 2.3: Different AGV kinematics employed in industrial contexts; (a): pure differential drive with pantograph lifter; (b): tricycle with fork lifter; (c): non-holonomic four wheel AGV with double fork lifter; (d): holonomic four wheel AGV with double conveyor.

Pure differential drive vehicles like the one in Figure 2.3a can lift loads from below exploiting a pantograph. Three wheel AGVs, also called *trycicles*, as the one in Figure 2.3b can lift and move materials using forks. To increase the flow of wares non-holonomic four wheel AGVs, like the one in Figure 2.3c, double the forks and therefore the load. Finally, holonomic four wheel AGVs as the one in Figure 2.3d couple lateral movements and conveyors to move materials.

Regarding non-holonomic four wheel AGVs, this thesis will focus on two particular kinematics:

- Forward steer and forward drive AGVs, usually called *Ackermann AGVs*;
- Forward steer and backward drive AGVs, known as *Dual Drive AGVs*.

Each of the previous kinematic is well known within the community of robotics and automation, so their models are widespread. Nevertheless, the kinematics of each vehicle embeds some parameters that must be calibrated to work properly.

All the aforementioned AGVs must be able to position correctly w.r.t. the whole fleet; accurate and repeatable positioning are crucial to lift and deploy materials successfully within the warehouse.

The mechanical measures of the chassis of each AGV are quite precise (regarding wheelbases, axis lengths, etc.) but other parameters as laser scanner pose, motor-wheels' steering offsets and drive multipliers must be correctly tuned.

As explained in Chapter 1, the typical manual calibration procedure is time consuming and error prone, while also requiring very experienced technicians. The novel automatic calibration method presented in Chapter 3 improves upon the manual method for being faster, more accurate and repeatable, in addition to provide safer AGV movements and positioning.

2.4 Sensing Technologies for AGV Navigation

AGVs are sometimes called LGVs (*Laser Guided Vehicles*) due to the laser rangefinder they are often equipped with. Industrial vehicles exploit the information provided by this sensor for localization, navigation and obstacle detection purposes. Sensor manufacturers provide a variety of different sensing solutions including RGB cameras, Time Of Flight (ToF) sensors and structured-light devices. The latter are often used to identify *Data Matrices* and *QR Codes* on the floor of the plant as global positioning references [8–13]. The specifications of each sensor often depend on its field of application and its price range. Furthermore, modern manufacturing techniques coupled with advanced embedded software enable regular sensor updates and frequent new device releases.

However, laser rangefinders are still the most used localization sensors on AGVs since they offer the best trade-off between precision, measuring resolution, scanning frequency, robustness and cost. Laser scanners are commonly based on the ToF calculation, but other principles like phase variation or triangulation exist. Two frequently used industrial laser rangefinders are shown in Figure 2.4.

(a)

(b)

Figure 2.4: Different laser rangefinders based on the Time Of Flight principle; (a) Sick Nav350; (b) Pepperl+Fuchs R2000 UHD.

These laser sensors retrieve distance measures with a field-of-view of 360°, fixing angular step and rotation frequency. The resulting information about the environment is matched with a static reference to ensure adequate localization performance. These laser scanners can effectively and accurately detect obstacles covered with reflective materials, which can be used as artificial landmarks. The canonical approach of industrial localization systems matches a group of reflectors with a static map to ensure accurate localization. A typical map of reflectors is shown in Figure 2.5.

The number of reflectors required in a plant grows quickly with its floorplan, thus increasing costs and deployment time. Furthermore, this solution is not flexible since many changes in the industrial environment may require to add, remove or re-map some artificial landmarks.

The novel methods presented in Chapter 4 achieve precise localization and positioning, exploiting only environmental information that is available regardless artificial landmark deployment. These localization methods for unstructured environments

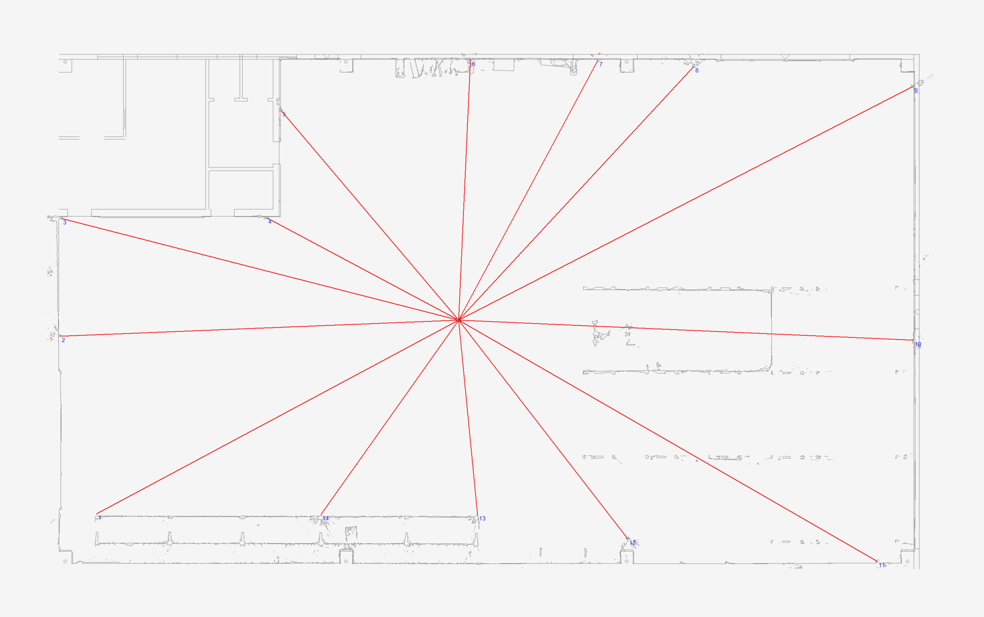


Figure 2.5: Artificial landmarks (small blue numbers) of a simple industrial warehouse detected from a hypothetical laser scanner in the middle.

rely on landmark maps (for feature matching) or dense maps (for 2D map matching). The dense map is obtained from the 3D point cloud generated by a *Terrestrial Laser Scanner* (TLS) while landmark maps are retrieved by a *Total Station* (TS). These sensors are displayed in Figure 2.6.

The aforementioned localization methods increase positioning precision, enhance flexibility and speed up the deployment of the automatized warehouse. In addition to that, 3D surveys are quite common during the early design phases of the plant. As a result, environmental information is already available and no artificial landmark must be added for localization and navigation.


(a) (b)

Figure 2.6: Measurement laser sensors used for environmental mapping; (a) Leica RTC360 Terrestrial Laser Scanner; (b) Leica TS60 Total Station.

2.5 Proposed Workflow for Automated Warehouse Design

The deployment of automated warehouses is a very complex task that requires careful planning and accurate design. To ensure fast installation times and satisfy customer's requirements, Elettric80 SpA has developed a standard workflow for automated warehouse design, shown in Figure 2.7. This Section will describe the canonical design workflow and then will discuss the improvements introduced in this dissertation.

In the standard approach, the first crucial phase after the commercial sale is the *3D TLS Survey*. During this step, an accurate 3D representation of the industrial environment is acquired, serving as ground-truth for warehouse design, machinery placement and obstruction detection [14–16]. Thereafter, during the *AGV Layout Design* phase, the pre-computed paths are created exploiting the information gathered during the TLS survey. Alongside, the reflectors are installed and mapped through the industrial environment during the *Reflector Measurement* phase.

When the AGV layout and the reflector map have been defined, the AGVs can be installed in the plant as shown in the *AGV Fleet Deployment* step. The set up of the fleet ends with the *Manual Calibration* of each AGV, that assures uniform behaviours of all the vehicles. The automated warehouse handled by AGVs becomes operational and the vehicles perform the *Reflector Localization*, exploiting the laser rangefinder and the landmark map previously defined [17]. Preventive maintenance interventions or other unattended events may lead to further re-calibrations of the vehicles.

The described workflow can be improved in the following aspects with the contributions proposed in this thesis, as follows:

- The *Automatic Calibration* step substitutes the manual procedure, reducing deployment times and boosting AGV positioning performance.
- The *Map Creation* step replaces the reflector measurement phase, exploiting the information from the 3D survey to create an accurate map of the environment that can be used for localization.
- The *Natural Localization* step replaces the localization with reflectors, employing the map of the environment and advanced localization methods.

These novel steps will be described in detail in the next chapters.

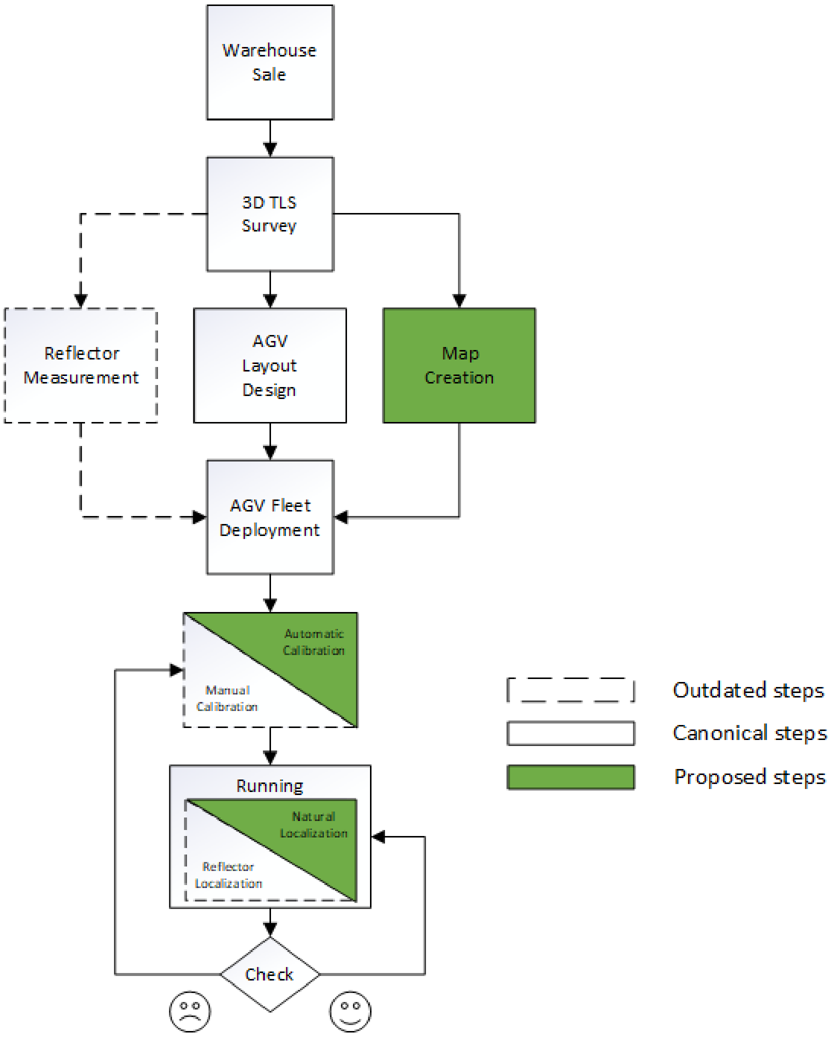


Figure 2.7: Proposed workflow for automated warehouse design; the *Proposed steps* are inserted in the already existent sequence of *Canonical steps*, substituting the *Outdated steps*.

Chapter 3

Automatic Calibration of Industrial AGVs

This chapter presents a novel method for extrinsic and intrinsic automatic calibration of four wheel industrial AGVs compliant with Ackermann and Dual Drive kinematics. The goal of the calibration procedure is to estimate a set of parameters with the aim of predicting the behaviour of the vehicle as accurately as possible, to improve localization. For each kinematic model the algorithm estimates the trajectories measured by an on-board sensor and the expected ones given the state of the wheels. The estimation exploits the model equations derived in this chapter which constrain calibration parameters and measurements from wheel encoders and sensor odometry. The parameter values are computed through closed-form solutions of least-square estimation.

The developed procedure computes the parameters in about $10 - 15$ minutes, a significant improvement compared with one hour or more required by manual AGV calibration. Experiments with AGVs of various sizes in a warehouse have assessed the accuracy and stability of the proposed approach. The position accuracy achieved by AGVs calibrated with the proposed method is higher than the one achieved by manual calibration.

3.1 Introduction

AGVs are commonly used to transport goods and to efficiently handle logistics of industrial warehouses. Industrial AGVs are usually equipped with forks or other grasping devices for transportation of goods, and with one or more exteroceptive sensors to detect obstacles and to estimate robot position and orientation. Each vehicle must be able to localize in the environment, to perform navigation, and to precisely reach the operation points with the forklift to lay down or carry pallets. Accurate robot positioning depends on accurate odometry and sensor-based localization, which in turn requires suitable calibration.

AGV calibration is the estimation of *intrinsic parameters*, which relate the wheel commands to the motion of the AGV, and *extrinsic parameters*, which define the pose of the sensor placed on the robot and used for localization [18]. Examples of intrinsic parameters include those used in the computation of odometry like the wheel steering offsets and the wheel driving scales relating encoders and travelled distance. Examples of extrinsic parameters include the relative position and orientation of the on-board sensors w.r.t. the robot frame. In industrial practice, calibration parameters are often evaluated by iteratively correcting parameter values until the desired motion is obtained. Such calibration process requires from half an hour to one hour and half for each AGV and its accuracy largely depends on the skills and experience of the operator performing the calibration. Consistency of manual calibration with multiple, possibly different AGVs operating in the same warehouse is often hard to guarantee, which can result in different behavior when reaching the operating points.

In this chapter, a novel algorithm for intrinsic and extrinsic calibration of four wheel AGVs, namely Ackermann and Dual Drive kinematic models, is presented. Four wheel AGVs are increasingly preferred in industrial applications over simpler kinematic configurations due to their better load distribution and stability on rough grounds, as well as because of their actuation redundancy and more precise fork operations. Figure 3.1 shows an example of Ackermann AGV. Both Ackermann and Dual Drive models have steering front wheels and non-steering rear wheels. In Ackermann AGVs the two front wheels are actuated, whereas in Dual Drive models the

Figure 3.1: An industrial Ackermann AGV with visible front steering and actuated wheels, its back forklift and a navigation laser scanner at the top of a telescopic pole. Although with slightly different mechanical structure, Dual Drive AGVs have a similar appearance.

rear wheels are actuated. In Dual Drive AGVs the actuators are closer to the fork-lift on the rear of vehicles, which makes these vehicles more effective in cargo operations. To date, no suitable automated calibration procedure has been proposed for four wheel AGVs. This work stems from a joint industry-academia project aiming at improving calibration accuracy and reducing calibration time through automatic procedures. Moreover, the developed solution for calibration must take into account the software limitations of standard embedded industrial controllers (e.g. programmable logic controllers) adopted in AGVs.

Research has addressed several formulations of calibration problems for different robotic systems, including the calibration of multi-sensor systems and robot odometry [18–51]. In industrial scenarios, calibration is required for the correct operation of wheel mobile robots [18, 20, 27, 29, 52, 53], multi-tractor vehicles with specific kinematics [51], and sensor-based object grasping using manipulators [39–42]. Several of these works investigate either intrinsic or extrinsic calibration: some techniques are focused on robot intrinsic kinematics [20, 21, 25–27, 29], whereas other are designed for multi-sensor systems [33, 39, 40, 42, 48–50].

Recently, Censi et al. [19] proposed a complete calibration algorithm for robots with differential drive kinematics and equipped with a sensor. However, the kinematic model, the selection of parameters and the presented experimental validation are mostly suited for a laboratory robotic platform rather than industrial vehicles. Most of the odometry calibration literature is committed to differential drive robots. The popularity of the differential drive actuation system is due to its simplicity. However, the industrial AGVs employed in transportation of heavy loads require robot kinematics with higher number of wheels.

A previous work [18] illustrated a calibration algorithm for tricycle robots, a configuration adopted in earlier design of industrial AGVs, but which is often replaced by four wheel AGVs in new automated warehouses. There are relatively few works on the calibration four wheel robots. McKerrow et al. [54] present a semi-automatic procedure computing driving scales and steering offsets, under the assumption that wheels are already aligned. The procedure proposed in [55] estimates step by step the pose of a range finder mounted on an ominidirectional robot using segment features, the gyroscope orientation and the odometric parameters.

None of these methods optimize all the calibration parameters of a four wheel AGV in a single step. Maye et al. [56] proposed a general calibration framework that can estimate all kind of parameters, intrinsic or extrinsic, provided a model. The main advantages of such method are its applicability for online calibration and the automatic numerical evaluation of observability. However, since it is designed for general AGV models, it is based on numerical methods and model approximations (e.g. linearization, covariance projection based on Jacobian, etc.). Moreover, online continous calibration may suffer from outlier measurements in the case of industrial AGVs with lifelong operability. In order to achieve the high position accuracy required in industrial applications, it would be convenient to evaluate all the calibration parameters of four wheel AGVs by optimizing an objective function depending on all the measurements. Unfortunately, four wheel models are over-actuated and intrinsic kinematic parameters cannot be identified by means of the observed trajectory.

In this chapter we develop a complete calibration procedure of four wheel AGVs according to the principle adopted in [18, 19]: the comparison of the trajectory measured by the on-board sensor and the expected trajectory of the AGV. Due to the observability issues of four wheel kinematic models, the only pre-condition is the manual alignment of the front wheels of the robot. To observe trajectories, the robot must be equipped with on-board sensors enabling egomotion estimation. Sensors with such capability, like range finders, are usually mounted on industrial AGVs and other mobile robots for localization and navigation as well as obstacle detection. The expected trajectories also depend on odometry and, thus, on intrinsic parameters. These different measurements are encoded by a set of constraints among intrinsic and extrinsic parameters, which are exploited to perform least-square estimation.

In the Ackermann and Dual Drive kinematic models addressed in this chapter, the presence of multiple independently actuated wheels makes calibration more difficult as well as crucial to limit slipping and to achieve consistent motion of all actuators. The main contribution of this chapter is the derivation of equations for four wheel Ackermann and Dual Drive kinematics, and their closed-form solutions. The intrinsic calibration equations accurately describe the real motion of AGVs under the assumption of wheel alignment and negligible wheel slipping.

The presented extrinsic calibration algorithm extends a previous work on tricycle AGVs [18]. Moreover, the proposed closed-form solution is suitable for implementation on PLCs (Programmable Logic Controllers) used for industrial AGVs. The second contribution is the implementation, deployment and assessment of the proposed methods on industrial AGVs. In particular, the implemented application enables full estimation of calibration parameters in about $10-15$ minutes. Repeated calibration trials have exhibited numerical stability and precision in the values of computed parameters. Positioning tests on actual AGVs have also demonstrated the higher accuracy of the proposed calibration algorithm w.r.t. manual calibration.

The chapter is organized as follows. Section 3.2 presents the general model of four wheel AGVs and the specific equations of Ackermann and Dual Drive robots. Sections 3.3 and 3.4 illustrate the solutions to respectively intrinsic and extrinsic calibration. Section 3.5 discusses practical implementation and deployment of calibration algorithms on industrial AGVs. Section 3.6 presents the results about calibration repeatability and position precision. Finally, the discussion in Section 3.7 summarize the chapter.

3.2 Problem Formulation

This Section illustrates the *Ackermann* and *Dual Drive* kinematics and provides the formulation of their intrinsic parameter estimation. These two models have four wheels and are both over-actuated vehicles, since there are more actuators than the system degrees-of-freedom. Thus, large wheel slipping may occur with arbitrary values of steering and speed. When wheel commands yield a consistent AGV motion, i.e. correspond to the kinematics of a rigid body, these input commands are called *consistent*. When the commands are not consistent, robot dynamics must be taken into account to compute the final motion. Since accurate modelling and estimation of torques, masses and friction is difficult and beyond the scope of the dissertation, the analysis is limited to kinematics.

3.2.1 Modelling Four Wheel AGVs

Figure 3.2 illustrates the general structure of four wheel robots with reference frames and dimensional parameters. All the reference frames are treated in the following as either tridimensional frames with parallel axes $\hat{\mathbf{z}}$ or planar frames. Reference frames include a global inertial frame $\{G\}$ fixed in the environment, the robot frame $\{V\}$, the frame $\{S\}$ attached to the navigation sensor, and the wheel frames $\{F_l\}$ (front-left wheel), $\{F_r\}$ (front-right wheel), $\{B_l\}$ (back left wheel), and $\{B_r\}$ (back-right wheel). The two front wheels $\{F_l\}$ and $\{F_r\}$ have independent steering actuators controlling the respective steering angles. The dimensional parameters are the so called wheelbase f_x, i.e. the distance between front and rear axes, and the front and back wheel half-axes, respectively f_y and b_y.

The Ackermann and Dual Drive models are compliant with this geometric description. In the Ackermann model, the front wheels F_l and F_r are both steering and

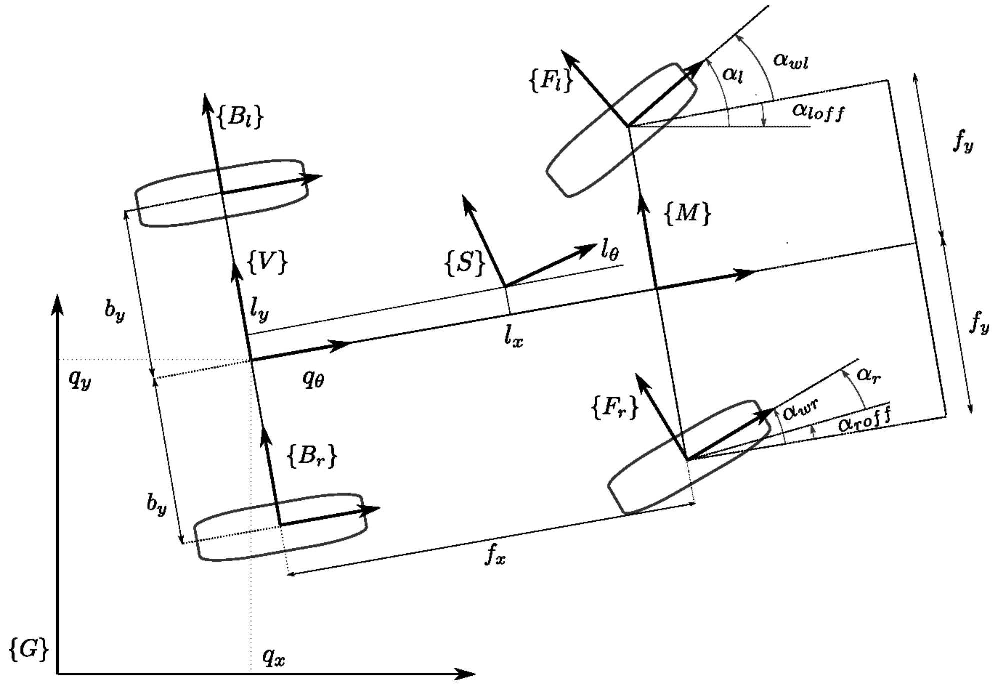


Figure 3.2: Structure of four wheel AGVs with the reference frames of AGV and its parameters. Counterclockwise arrows refer to positive angles, whereas clockwise ones (e.g. the arrow of α_{loff}) are negative.

actuated whereas the rear wheels B_l and B_r are passive. The rolling speed of passive wheels is completely set by the other actuated wheels. In the Dual Drive model, F_l and F_r are only steering whereas B_l and B_r are actuated and not steering. The front steering-only wheels are used to guide the vehicle and to limit slipping.

Since all the frames belong to a rigid body, the robot configuration in planar space can be described by a unique pose vector $\mathbf{q} = [q_x, q_y, q_\theta]^\top \in \mathrm{se}(2)$. The state vector represents the position $\mathbf{q}_{pos} = [q_x, q_y]^\top \in \mathrm{R}^2$ and orientation $q_\theta \in \mathrm{so}(2)$ of reference frame $\{V\}$ w.r.t. the global frame $\{G\}$. Henceafter, the subscript $\cdot_{pos}$ applied to a pose vector refers to its position coordinate vector in $\mathbb{R}^2$. The reference frame $\{V\}$ is sometimes called logical point, since robot state is logically represented as a point particle located in the origin of this frame. The general kinematic equation describing the evolution of state vector $\mathbf{q}$ over time is

$$\dot{\mathbf{q}} = \begin{bmatrix} v_{lp} \cos(q_\theta) \\ v_{lp} \sin(q_\theta) \\ \omega_{lp} \end{bmatrix} \tag{3.1}$$

The input controls of Equation (3.1) are given by the linear and angular velocities v_{lp} and ω_{lp} of the logical point. It is sometimes convenient to express the linear and angular speeds of robot frame $\{V\}$ from global frame $\{G\}$ in $\{V\}$ 3D coordinates as

$$^{V}\mathbf{v}_{V,G} = \begin{bmatrix} v_{lp} & 0 & 0 \end{bmatrix}^\top \tag{3.2}$$

$$^{V}\boldsymbol{\omega}_{V,G} = \begin{bmatrix} 0 & 0 & \omega_{lp} \end{bmatrix}^\top \tag{3.3}$$

Observe that $^{V}\mathbf{v}_{V,G}$ has only a longitudinal component aligned with axis $\hat{\mathbf{x}}_V$ of frame $\{V\}$, and zero lateral velocity. Odometry is obtained by integration of Equation (3.1) over a given time interval. The exact expression is straightforwardly obtained under the assumption that input controls v_{lp} and ω_{lp} are constant over $[t_{k-1}, t_k]$. Thus, the curvature radius $r_\kappa^k = v_{lp}/\omega_{lp}$ is also constant and the robot performs a circular path. The analytical expression of robot relative motion $\boldsymbol{\delta}^k = [\boldsymbol{\delta}_{pos}^{k\ \top}, \delta_\theta^k]^\top = [\delta_x^k, \delta_y^k, \delta_\theta^k]^\top \in$

se(2) is given by

$$\begin{aligned}\boldsymbol{\delta}_{pos}^{k} &= \mathbf{R}\left(-q_{\theta}(t_{k-1})\right)\ \left(\mathbf{q}_{pos}(t_{k}) - \mathbf{q}_{pos}(t_{k-1})\right) \\ &= r_{\kappa}^{k}\left[\begin{array}{c}\sin\left(\delta_{\theta}^{k}\right) \\ 1-\cos\left(\delta_{\theta}^{k}\right)\end{array}\right]\end{aligned} \tag{3.4}$$

$$\delta_{\theta}^{k} = q_{\theta}(t_{k}) - q_{\theta}(t_{k-1}) = \int_{t_{k-1}}^{t_{k}} \omega_{lp}\, d\tau \tag{3.5}$$

Equations (3.4) and (3.5) provide the exact expression of the relative motion over circular path segments.

Physically, the values of the velocities v_{lp} and ω_{lp} of the robot logical point depend on the robot wheels. The forward kinematic model consists of the equations relating the wheel steering angles and rolling speeds to the logical point velocities. The parameters used in forward kinematic model are called *intrinsic parameters*. In Section 3.2.2, the forward kinematic models and the intrinsic parameters of Ackermann and Dual Drive models is illustrated.

The robot is equipped with a sensor, which allows navigation and egomotion estimation. In the case of industrial AGVs, this sensor is usually a planar sensor like a laser scanner specifically designed to localize the robot in the environment by detecting artificial landmarks. The range finder mounted on the robot is such that the scanning plane is parallel to the ground plane. Thus, the position and orientation of sensor frame $\{S\}$ w.r.t. the robot frame $\{V\}$ is described by the planar pose vector $\mathbf{l} = [l_x, l_y, l_\theta]^\top \in \mathrm{se}(2)$. The parameters $\mathbf{l}$ that encode the sensor pose w.r.t. to the vehicle are called *extrinsic parameters*. Accurate estimation of extrinsic parameters is needed to perform robot localization and navigation using the sensor. The measurements acquired from the sensor enable the estimation of relative motion $\boldsymbol{\zeta}^k = [\zeta_x^k, \zeta_y^k, \zeta_\theta^k]^\top \in$ se(2) of frame $\{S\}$ over time interval $[t_{k-1}, t_k]$. In the case of laser scanner, the value $\boldsymbol{\zeta}^k$ is computed using registration algorithms [57].

The simultaneous estimation of intrinsic and extrinsic calibration parameters is performed through the comparison of the odometry $\boldsymbol{\delta}^k$ and the sensor egomotion $\boldsymbol{\zeta}^k$ over several path segments $k = 1, \ldots, n$. The value of $\boldsymbol{\delta}^k$ depends on the intrinsic parameters of the robot through forward kinematics as illustrated in Section 3.2.2.

On the other hand, $\boldsymbol{\delta}^k$ and $\boldsymbol{\zeta}^k$ are constrained by the extrinsic parameters $\mathbf{l}$ through the following relation

$$\begin{bmatrix} \mathbf{l}_{pos} + \mathbf{R}\left(l_\theta\right) \boldsymbol{\zeta}^k_{pos} \\ l_\theta + \zeta^k_\theta \end{bmatrix} = \begin{bmatrix} \boldsymbol{\delta}^k_{pos} + \mathbf{R}\left(\delta^k_\theta\right) \mathbf{l}_{pos} \\ \delta^k_\theta + l_\theta \end{bmatrix} \tag{3.6}$$

The symbol $\mathbf{R}(\cdot)$ refers, henceafter, both to the map from an angle $\beta \in S^1$ to the corresponding rotation matrix of $\mathbf{R}(\beta) \in \mathrm{SO}(2)$ and to the map from a vector $b \in \mathbb{R}^2$ to a skew matrix $\mathbf{R}(b) \in \mathbb{R}^{2\times 2}$ defined respectively as

$$\mathbf{R}(\beta) \triangleq \begin{bmatrix} \cos\beta & -\sin\beta \\ \sin\beta & \cos\beta \end{bmatrix}, \mathbf{R}(b) \triangleq \begin{bmatrix} b_x & -b_y \\ b_y & b_x \end{bmatrix} \tag{3.7}$$

The angular part of Equation (3.6) implies that the relative rotation angles of the robot and of the sensor attached as a rigid body are equal: $\delta^k_\theta = \zeta^k_\theta$. Since forward kinematics can relate the input control and intrinsic parameters to each δ^k_θ, this equation will be used to solve intrinsic calibration. The position part of Equation (3.6) constrains $\boldsymbol{\delta}^k_{pos}$ (computed from controls and intrinsic parameters) and $\boldsymbol{\zeta}^k_{pos}$ (directly measured using sensor registration) with the extrinsic parameters $\mathbf{l}$. The difference between the two members of Equation (3.6) can be interpreted as an error vector $\mathbf{e}_k$ dependent from $\mathbf{l}$. Thus, the value of $\mathbf{l}$ can be computed by least-square estimation over errors $\mathbf{e}_k$ for all path segments $k = 1, \ldots, n$.

3.2.2 Forward Kinematics of Four Wheel AGVs

The velocities of logical points are related to the velocities set by the wheels. The relations between speeds can be computed under the hypothesis that the robot can be treated as a rigid body. In particular, we can use the *wheel equation* [58] applied to each wheel frame $\{W\}$

$${}^V\mathbf{v}_{W,G} = {}^V\mathbf{v}_{V,G} + {}^V\boldsymbol{\omega}_{V,G} \times {}^V\mathbf{p}_W \tag{3.8}$$

where W is either F_l, F_r, B_l or B_r. The resulting velocities at the four wheels in robot coordinates are obtained through the substitution of Equation (3.2), (3.3) and the position vector of each wheel (see the parameters in Figure 3.2) in Equation (3.8) as

$$ {}^{V}\mathbf{v}_{F_l,G} = \begin{bmatrix} v_{lp} - \omega_{lp} f_y & \omega_{lp} f_x & 0 \end{bmatrix}^{\top} \tag{3.9} $$

$$ {}^{V}\mathbf{v}_{F_r,G} = \begin{bmatrix} v_{lp} + \omega_{lp} f_y & \omega_{lp} f_x & 0 \end{bmatrix}^{\top} \tag{3.10} $$

$$ {}^{V}\mathbf{v}_{B_l,G} = \begin{bmatrix} v_{lp} - \omega_{lp} b_y & 0 & 0 \end{bmatrix}^{\top} \tag{3.11} $$

$$ {}^{V}\mathbf{v}_{B_r,G} = \begin{bmatrix} v_{lp} + \omega_{lp} b_y & 0 & 0 \end{bmatrix}^{\top} \tag{3.12} $$

The above equations provide the velocity required for each wheel in order to achieve the desired velocities v_{lp} and ω_{lp} in the logic point.

However, each actuated wheel is controlled by its rolling speed and, for steering wheels, by its steering angle through actuators. We describe the parameters that enable setting of wheel velocities referred to the four wheel robot illustrated in Figure 3.2.

- *Steering offsets*. The two front wheels are steering and their steering angles α_{wl} and α_{wr} are referred to the forward direction of robot corresponding to axis $\hat{\mathbf{x}}_V$ of frame $\{V\}$. The steering motors control the steering angles and their encoders measure the angles α_l and α_r w.r.t. the steer encoder reference. Ideally, the steer encoder references should be aligned with the forward direction of robot, i.e. axis $\hat{\mathbf{x}}_V$ of frame $\{V\}$. In practice, there are *steering offsets* α_{loff} and α_{roff} of encoders such that

$$ \alpha_{wl} = \alpha_l + \alpha_{loff} \qquad\qquad \alpha_{wr} = \alpha_r + \alpha_{roff} \tag{3.13} $$

- *Driving scales*. The distance travelled by an actuated wheel is proportional to its turning angle. Physically, the turning angle over a time interval $[t_{k-1}, t_k]$ is measured by counting the corresponding tick number of the wheel encoder n_w^k. The driving scale s_w is the parameter that relates the relative encoder tick n_w^k to the travelled distance $s_w n_w^k$ (e.g. in *mm*). The corresponding wheel velocity at time t is denoted as $s_w \dot{n}_w(t)$. In the Ackermann model, the left and right front wheels are actuated and the module of their velocities is equal respectively to

$$ \|{}^{V}\mathbf{v}_{F_l,G}\| = s_{wl}\dot{n}_{wl} \qquad\qquad \|{}^{V}\mathbf{v}_{F_r,G}\| = s_{wr}\dot{n}_{wr} \tag{3.14} $$

where s_{wl} and s_{wr} represent respectively the left and right driving scales, and $\dot{n}_{wl}$ and $\dot{n}_{wl}$ are the turning rates of left and right wheels measured in encoder ticks. In the Dual Drive model, the same notation is adopted for the rear wheels

$$\|{}^{V}\mathbf{v}_{B_l,G}\| = s_{wl}\dot{n}_{wl} \qquad \|{}^{V}\mathbf{v}_{B_r,G}\| = s_{wr}\dot{n}_{wr} \tag{3.15}$$

where s_{wl}, s_{wr}, $\dot{n}_{wl}$ and $\dot{n}_{wl}$ refer to the rear wheels. The physical meaning of these quantities, i.e. whether referred to front or back wheels, will be clear from the context.

Forward kinematics consist of the equations relating the logical point velocities to the wheel controls. Forward kinematics depend on the *intrinsic parameters* α_{loff}, α_{roff}, s_{wl} and s_{wr} discussed above. The goal of intrinsic calibration is the estimation of such parameters. We derive the equations that relate the velocities of the robot logical point and the velocities of the wheels for Ackermann and Dual Drive models.

Ackermann model. In the Ackermann model, the two front wheels are steering and actuated, whereas the rear wheels are passive and not controlled. Its forward kinematics is obtained by comparing the velocities of frames $\{F_l\}$ and $\{F_r\}$, from Equation (3.9)-(3.10), with the wheel velocities in Equation (3.14) as

$${}^{V}\mathbf{v}_{F_l,G} = \begin{bmatrix} s_{wl}\,\dot{n}_{wl}\,\mathrm{c}\,\alpha_{wl} \\ s_{wl}\,\dot{n}_{wl}\,\mathrm{s}\,\alpha_{wl} \\ 0 \end{bmatrix} = \begin{bmatrix} v_{lp} - \omega_{lp} f_y \\ \omega_{lp} f_x \\ 0 \end{bmatrix} \tag{3.16}$$

$${}^{V}\mathbf{v}_{F_r,G} = \begin{bmatrix} s_{wr}\,\dot{n}_{wr}\,\mathrm{c}\,\alpha_{wr} \\ s_{wr}\,\dot{n}_{wr}\,\mathrm{s}\,\alpha_{wr} \\ 0 \end{bmatrix} = \begin{bmatrix} v_{lp} + \omega_{lp} f_y \\ \omega_{lp} f_x \\ 0 \end{bmatrix} \tag{3.17}$$

where the modules of wheel velocities in Equation (3.14) are projected according to steering angles α_{wl} and α_{wr}. Forward kinematics of Ackermann model is given by

$$v_{lp} = \frac{s_{wl}\dot{n}_{wl}\,\mathrm{c}\,\alpha_{wl} + s_{wr}\dot{n}_{wr}\,\mathrm{c}\,\alpha_{wr}}{2} \tag{3.18}$$

$$\omega_{lp} = \frac{s_{wl}\;\dot{n}_{wl}\,\mathrm{s}\,\alpha_{wl}}{f_x} \tag{3.19}$$

$$\omega_{lp} = \frac{s_{wr}\;\dot{n}_{wr}\,\mathrm{s}\,\alpha_{wr}}{f_x} \tag{3.20}$$

$$\omega_{lp} = \frac{s_{wr}\;\dot{n}_{wr}\,\mathrm{c}\,\alpha_{wr} - s_{wl}\;\dot{n}_{wl}\,\mathrm{c}\,\alpha_{wl}}{2f_y} \tag{3.21}$$

Since Ackermann AGVs are over-actuated, there are multiple expressions for ω_{lp}. The values of independent controls α_l, α_r, $\dot{n}_{wl}$ and $\dot{n}_{wr}$ must be chosen s.t. Equations (3.18)-(3.21) are simultaneously satisfied. Otherwise, kinematics is insufficient to model wheel slipping and system dynamics must be taken into account to predict its evolution.

Dual Drive model. Dual Drive AGVs have two front steering wheels and two rear actuated wheels, which cannot steer. The same procedure adopted for the Ackermann model can be applied to compute Dual Drive forward kinematics. The velocities ${}^{V}\mathbf{v}_{B_l,G}$ and ${}^{V}\mathbf{v}_{B_r,G}$ of rear wheels have only longitudinal components along axis $\hat{\mathbf{x}}_V$, that are equal to

$$s_{wl}\;\dot{n}_{wl} = v_{lp} - \omega_{lp} b_y \tag{3.22}$$

$$s_{wr}\;\dot{n}_{wr} = v_{lp} + \omega_{lp} b_y \tag{3.23}$$

Such equations can be used to compute the forward kinematics of Dual Drive

$$v_{lp} = \frac{s_{wl}\;\dot{n}_{wl} + s_{wr}\;\dot{n}_{wr}}{2} \tag{3.24}$$

$$\omega_{lp} = \frac{s_{wr}\;\dot{n}_{wr} - s_{wl}\;\dot{n}_{wl}}{2b_y} \tag{3.25}$$

Driving scales s_{wl} and s_{wr} are the only intrinsic parameters appearing in the above relations. The steering angles are required for the steering passive wheels described by

$$
{}^{V}\mathbf{v}_{F_l,G} = \begin{bmatrix} v_{fl}\,\mathrm{c}\,\alpha_{wl} \\ v_{fl}\,\mathrm{s}\,\alpha_{wl} \\ 0 \end{bmatrix} = \begin{bmatrix} v_{lp} - \omega_{lp} f_y \\ \omega_{lp} f_x \\ 0 \end{bmatrix} \tag{3.26}
$$

$$
{}^{V}\mathbf{v}_{F_r,G} = \begin{bmatrix} v_{fr}\,\mathrm{c}\,\alpha_{wr} \\ v_{fr}\,\mathrm{s}\,\alpha_{wr} \\ 0 \end{bmatrix} = \begin{bmatrix} v_{lp} + \omega_{lp} f_y \\ \omega_{lp} f_x \\ 0 \end{bmatrix} \tag{3.27}
$$

In Equations (3.26) and (3.27) the modules of wheel velocities, respectively v_{fl} and v_{fr} for front left and right wheels, are not set by motor commands. These equations can be manipulated to write two constraints, one for each front wheel, according to the following procedure (described for the front left wheel, but similar for the right one). We can remove the dependency from v_{fl} by multiplying the x and y component of Equation (3.26) by respectively $\mathrm{s}\,\alpha_{wl}$ and $\mathrm{c}\,\alpha_{wl}$. Then, the two equations are subtracted member by member. Finally, the values of v_{lp} and ω_{lp} from Equations (3.24) and (3.25) can be substituted. The final result is the following:

$$
\begin{aligned}
0 =& \frac{s_{wl}\,\dot{n}_{wl} + s_{wr}\,\dot{n}_{wr}}{2}\,\mathrm{s}\,\alpha_{wl} \\
&+ \frac{s_{wr}\,\dot{n}_{wr} - s_{wl}\,\dot{n}_{wl}}{2b_y}\,(-f_y\,\mathrm{s}\,\alpha_{wl} - f_x\,\mathrm{c}\,\alpha_{wl}) \qquad (3.28) \\
0 =& \frac{s_{wl}\,\dot{n}_{wl} + s_{wr}\,\dot{n}_{wr}}{2}\,\mathrm{s}\,\alpha_{wr} \\
&+ \frac{s_{wr}\,\dot{n}_{wr} - s_{wl}\,\dot{n}_{wl}}{2b_y}\,(f_y\,\mathrm{s}\,\alpha_{wr} - f_x\,\mathrm{c}\,\alpha_{wr}) \qquad (3.29)
\end{aligned}
$$

3.2.3 Discussion

The model presented in the previous Subsection is based on the assumption that the velocity directions and steering angles of wheels are congruent. In particular, Equations (3.16)-(3.17) for Ackermann and (3.26)-(3.27) for Dual Drive state that vectors ${}^{V}\mathbf{v}_{F_l,G}$ and ${}^{V}\mathbf{v}_{F_r,G}$ have orientation angles α_{wl} and α_{wr}. This assumption holds only when no wheel slippage occurs and the input controls are consistent with the rigid body motion. Otherwise, since Ackermann and Dual Drive are over-actuated kinematics, the kinematic parameters cannot be identified by means of the observed trajectory. For example, the same straight motion can be obtained either with perfectly parallel wheels or with symmetrically opposite steering angles. Thus, before the calibration procedure, a manual parallelization of wheels is performed to guarantee the feasibility of such condition. The procedure is briefly described in Section 3.5.

The wheel parallelization procedure is performed with tolerance on offset accuracy. Once the wheel are approximately aligned, the optimization methods for the two four wheel kinematics described in Sections 3.3 and 3.4 can effectively estimate the calibration parameters, as shown by experiments.

3.3 Intrinsic Calibration

The intrinsic calibration procedures of Ackermann and Dual Drive models depend on their specific equations. Nonetheless, the principle applied in intrinsic calibration is the same. Since the AGV is a rigid body, the rotation angles of all the frames rigidly attached to the robot are the same. In order to use the simplified Equations of odometry (3.4)-(3.5), only time intervals $[t_{k-1}, t_k]$ with constant control input are taken into account. The ^{k} apex notation is used henceafter for all the parameter values and the measurements related to time intervals $[t_{k-1}, t_k]$ with $k = 1, \dots, n_k$. The path travelled over $[t_{k-1}, t_k]$ with constant wheel steering angles and velocities is called *path segment k*. In particular, for each segment k, we measure the following variables:

- the steering angles of left and right front wheels, respectively α_l^k and α_r^k, which are constant on path segment k and measured by the steering encoders;
- the left and right wheel spins n_{wl}^k and n_{wr}^k measured in encoder ticks (front wheels for Ackermann, rear wheels for Dual Drive);
- the relative rotation δ_θ^k of the AGV while following a path segment between time instant t_{k-1} and t_k.

As observed before, the relative rotation δ_θ^k is not measured directly, but obtained through sensor registration ζ_θ^k. In the industrial setup, sensor registration is performed by matching artificial landmarks made of reflective material in known positions. These landmarks are called *reflectors* and are detected by the laser scanner.

3.3.1 Calibration of Ackermann Model

The solution of Ackermann intrinsic calibration can be obtained by integrating Equations (3.19)-(3.21), which relate angular velocity ω_{lp} to wheel parameters, over several path segments k. The three expressions of ω_{lp} can be substituted into Equation (3.5) and equated to $\delta_\theta^k = \zeta_\theta^k$. Moreover, we substitute the expression of total steering angles $\alpha_{wl}^k = \alpha_l^k + \alpha_{loff}$ and $\alpha_{wr}^k = \alpha_r^k + \alpha_{roff}$ and of the wheel spin angles

$$n_{wl}^k = \int_{t_{k-1}}^{t_k} \dot{n}_{wl} d\tau \quad n_{wr}^k = \int_{t_{k-1}}^{t_k} \dot{n}_{wr} d\tau \tag{3.30}$$

The result of this operation over path segment k consists of the three linear equations

$$\mathbf{A}_a^k \, \mathbf{x}_a = \mathbf{b}_a^k \tag{3.31}$$

where the matrices and vectors are defined as

$$\mathbf{x}_a = \begin{bmatrix} x_{a1} \\ x_{a2} \\ x_{a3} \\ x_{a4} \end{bmatrix}^\top = \begin{bmatrix} s_{wl}\,\mathrm{c}\,\alpha_{loff} \\ s_{wl}\,\mathrm{s}\,\alpha_{loff} \\ s_{wr}\,\mathrm{c}\,\alpha_{roff} \\ s_{wr}\,\mathrm{s}\,\alpha_{roff} \end{bmatrix} \tag{3.32}$$

$$\begin{aligned} \mathbf{A}_a^k &= \begin{bmatrix} \frac{n_{wl}^k\,\mathrm{s}\,\alpha_l^k}{f_x} & \frac{n_{wl}^k\,\mathrm{c}\,\alpha_l^k}{f_x} & 0 & 0 \\ 0 & 0 & \frac{n_{wr}^k\,\mathrm{s}\,\alpha_r^k}{f_x} & \frac{n_{wr}^k\,\mathrm{c}\,\alpha_r^k}{f_x} \\ \frac{-n_{wl}^k\,\mathrm{c}\,\alpha_l^k}{2f_y} & \frac{n_{wl}^k\,\mathrm{s}\,\alpha_l^k}{2f_y} & \frac{n_{wr}^k\,\mathrm{c}\,\alpha_r^k}{2f_y} & \frac{-n_{wr}^k\,\mathrm{s}\,\alpha_r^k}{2f_y} \end{bmatrix} \\ &= \begin{bmatrix} a_{slcl} & a_{slsl} & 0 & 0 \\ 0 & 0 & a_{srcr} & a_{srsr} \\ a_{clcl} & a_{clsl} & a_{crcr} & a_{crsr} \end{bmatrix} \end{aligned} \tag{3.33}$$

$$\mathbf{b}_a^k = \begin{bmatrix} \delta_\theta^k & \delta_\theta^k & \delta_\theta^k \end{bmatrix}^\top \tag{3.34}$$

The original intrinsic parameters s_{wl}, s_{wlr}, α_{loff} and α_{roff} have been conveniently substituted with other variables s.t. the equations are linear w.r.t. the new variables. Of course, additional constraints may be inserted to keep consistency, when the new variable space is a specific manifold. The above constraints are replicated for all path segments with $k = 1, \dots, n_k$ to form the linear system with matrix $\mathbf{A}_y$ and known term vector $\mathbf{b}_y$ defined as

$$\mathbf{A}_a = \begin{bmatrix} \mathbf{A}_a^{1\top} & \mathbf{A}_a^{2\top} & \dots & \mathbf{A}_a^{n_k\top} \end{bmatrix}^\top \tag{3.35}$$

$$\mathbf{b}_a = \begin{bmatrix} \mathbf{b}_a^{1\top} & \mathbf{b}_a^{2\top} & \dots & \mathbf{b}_a^{n_k\top} \end{bmatrix}^\top \tag{3.36}$$

A least-square solution $\mathbf{x}_a^*$ for this system can be found using *Moore-Penrose pseudoinverse* [59]. Thus, Ackermann model calibration can be formulated as the following problem:

Problem 1 (Ackermann intrinsic calibration)**.** *Find the parameter vector* $\mathbf{x}_a^* \in \mathbb{R}^4$ *that*

$$\text{minimize}||\mathbf{A}_a\mathbf{x}_a - \mathbf{b}_a||^2 \tag{3.37}$$

where $\mathbf{A}_a$ *and* $\mathbf{b}_a$ *are given respectively by Equation (3.35) and (3.36). The final calibration parameters are computed as*

$$s_{wl}^* = \sqrt{x_{a1}^2 + x_{a2}^2} \tag{3.38}$$

$$s_{wr}^* = \sqrt{x_{a3}^2 + x_{a4}^2} \tag{3.39}$$

$$\alpha_{loff}^* = \operatorname{atan2}(x_{a2}, x_{a1}) \tag{3.40}$$

$$\alpha_{roff}^* = \operatorname{atan2}(x_{a4}, x_{a3}) \tag{3.41}$$

3.3.2 Calibration of Dual-Drive Model

The solution of intrinsic calibration of Dual Drive AGVs is derived with a procedure analogous to the one applied to the Ackermann model. Yet there is an important difference between the formula of ω_{lp} in Equation (3.25) and the constraints on steering wheels in Equation (3.28)-(3.29). The first constraint can be used to estimate the driving scales s_{wl} and s_{wr} of respectively left and right back wheels. The latter relations are used to estimate the steering offsets of left and right front wheels, respectively α_{loff} and α_{roff}. This two-step estimation (first the estimation of driving scales, followed by steering offsets) has a closed-form solution, which is suitable for implementation in embedded systems. The integration of Equation (3.25) leads to

$$\delta_\theta^k = \underbrace{\left(\frac{n_{wr}^k}{2b_y}\right)}_{a_{\omega r}^k} s_{wr} - \underbrace{\left(\frac{n_{wr}^k}{2b_y}\right)}_{a_{\omega l}^k} s_{wl} \tag{3.42}$$

where the travelled angles n_{wl}^k and n_{wr}^k are measured by the left and right wheel encoders as in Section 3.3.1. The multiple constraints in the form of Equation (3.42) collected over path segments $k = 1, \ldots, n_k$ can be arranged into the linear system

$$\underbrace{\begin{bmatrix} \delta_\theta^1 \\ \delta_\theta^2 \\ \vdots \\ \delta_\theta^{n_k} \end{bmatrix}}_{\mathbf{b}_s} = \underbrace{\begin{bmatrix} a_{\omega l}^1 & a_{\omega r}^1 \\ a_{\omega l}^2 & a_{\omega r}^2 \\ \vdots & \vdots \\ a_{\omega l}^{n_k} & a_{\omega r}^{n_k} \end{bmatrix}}_{\mathbf{A}_s} \underbrace{\begin{bmatrix} s_{wl} \\ s_{wr} \end{bmatrix}}_{\mathbf{x}_s} \tag{3.43}$$

Similarly to previous case, the conditions given by the linear system in Equation (3.43) often do not hold simultaneously. Hence, the driving scales $\mathbf{x}_s$ are evaluated by minimizing $||\mathbf{A}_s\mathbf{x}_s - \mathbf{b}_s||$. Once the driving scales are evaluated, Equations (3.28) and (3.29) can be integrated over all path segments k. The left steering offset α_{loff} appears only in Equation (3.28), while α_{roff} only in Equation (3.29). Thus, two sets of independent equations in α_{loff} and α_{roff}, which can be solved independently, are obtained from the integration

$$\underbrace{\begin{bmatrix} a_{\alpha l,1}^1 & a_{\alpha,l,2}^1 \\ \vdots & \vdots \\ a_{\alpha l,1}^{n_k} & a_{\alpha,l,2}^{n_k} \end{bmatrix}}_{\mathbf{A}_{\alpha,l}} \cdot \underbrace{\begin{bmatrix} \mathrm{c}\,\alpha_{loff} \\ \mathrm{s}\,\alpha_{loff} \end{bmatrix}}_{\mathbf{x}_l=[x_{l1},x_{l2}]]^\top} = \mathbf{0} \tag{3.44}$$

$$\underbrace{\begin{bmatrix} a_{\alpha r,1}^1 & a_{\alpha,r,2}^1 \\ \vdots & \vdots \\ a_{\alpha r,1}^{n_k} & a_{\alpha,r,2}^{n_k} \end{bmatrix}}_{\mathbf{A}_{\alpha,r}} \cdot \underbrace{\begin{bmatrix} \mathrm{c}\,\alpha_{roff} \\ \mathrm{s}\,\alpha_{roff} \end{bmatrix}}_{\mathbf{x}_r=[x_{r1},x_{r2}]]^\top} = \mathbf{0} \tag{3.45}$$

where the coefficients of the two linear system matrices are defined as

$$a_v^k = \frac{s_{wl} n_{wl}^k + s_{wr} n_{wr}^k}{2} \tag{3.46}$$

$$a_\omega^k = \frac{s_{wr} n_{wr}^k - s_{wl} n_{wl}^k}{2 b_y} \tag{3.47}$$

$$a_{\alpha l,1}^k = (a_v^k - f_y a_\omega^k)\,\mathrm{s}\,\alpha_l^k - f_x a_\omega^k\,\mathrm{c}\,\alpha_l^k \tag{3.48}$$

$$a_{\alpha l,2}^k = (a_v^k - f_y a_\omega^k)\,\mathrm{c}\,\alpha_l^k + f_x a_\omega^k\,\mathrm{s}\,\alpha_l^k \tag{3.49}$$

$$a_{\alpha r,1}^k = (a_v^k - f_y a_\omega^k)\,\mathrm{s}\,\alpha_l^k - f_x a_\omega^k\,\mathrm{c}\,\alpha_l^k \tag{3.50}$$

$$a_{\alpha r,2}^k = (a_v^k - f_y a_\omega^k)\,\mathrm{c}\,\alpha_l^k + f_x a_\omega^k\,\mathrm{s}\,\alpha_l^k \tag{3.51}$$

Equations (3.44) and (3.45) have been written to highlight the linearity w.r.t. respectively vectors $\mathbf{x}_l$ and $\mathbf{x}_r$. The two variables are subject to constraints $||\mathbf{x}_l||^2 = 1$ and $||\mathbf{x}_r||^2 = 1$. Due to noise in measurements, the above conditions are usually not satisfied. The estimation of left and right steering offsets is achieved by finding the $\mathbf{x}_l$ or $\mathbf{x}_r$ that minimize respectively $||\mathbf{A}_{\alpha,l}\mathbf{x}_l||^2$ and $||\mathbf{A}_{\alpha,r}\mathbf{x}_r||^2$. Fortunately, the problem of minimizing a homogeneous quadratic function over a sphere is well-known and has a closed-form solution (see [60, p. 593]). It suffices to find the eigenvector of matrix $\mathbf{A}_{\alpha,\cdot}^\top \mathbf{A}_{\alpha,\cdot}$ corresponding to its minimum eigenvalue. The intrinsic calibration of Dual Drive model is summarized by the following problem:

Problem 2 (Dual Drive intrinsic calibration)**.** *Solve the following steps.*

1. *Find the driving scale vector* $\mathbf{x}_s^* = [s_{wl}, s_{wr}] \in \mathbb{R}^2$ *that*

$$\text{minimize} ||\mathbf{A}_s \mathbf{x}_s - \mathbf{b}_s||^2 \tag{3.52}$$

$$\text{s.t.} \quad y_{s1}, y_{s2} > 0 \tag{3.53}$$

where $\mathbf{A}_s$ *and* $\mathbf{b}_s$ *are defined in Equation (3.43).*

2. *After substitution of* $\mathbf{x}_s^*$*, find* $\mathbf{x}_l^*, \mathbf{x}_r^* \in \mathbb{R}^2$

$$\mathbf{x}_l^* = \underset{\|\mathbf{x}_l\|^2=1}{\operatorname{argmin}} \|\mathbf{A}_{\alpha,l}\mathbf{x}_l\|^2 \tag{3.54}$$

$$\mathbf{x}_r^* = \underset{\|\mathbf{x}_r\|^2=1}{\operatorname{argmin}} \|\mathbf{A}_{\alpha,r}\mathbf{x}_r\|^2 \tag{3.55}$$

The values of steering offsets *are computed as*

$$\alpha_{loff}^* = \operatorname{atan2}(x_{l2}^*, x_{l1}^*) \tag{3.56}$$

$$\alpha_{roff}^* = \operatorname{atan2}(x_{r2}^*, x_{r1}^*) \tag{3.57}$$

3.4 Extrinsic Calibration

The aim of extrinsic calibration is the estimation of the position and orientation of the sensor mounted on the robot represented by vector **l**. As illustrated in [18], extrinsic calibration is formulated as a least-square problem over the constraints derived from Equation (3.6). In particular, the position error on path segment k can be defined as

$$\begin{aligned} \mathbf{e}_{pos}^k &= (\mathbf{l}_{pos} + \mathbf{R}(l_\theta)\,\boldsymbol{\zeta}_{pos}^k) - \left(\boldsymbol{\delta}_{pos}^k + \mathbf{R}\left(\delta_\theta^k\right)\mathbf{l}_{pos}\right) \\ &= \underbrace{\begin{bmatrix} \mathbf{I}_2 - \mathbf{R}(\delta_\theta^k) & \mathbf{R}\left(\boldsymbol{\zeta}_{pos}^k\right) \end{bmatrix}}_{\mathbf{Q}_k} \underbrace{\begin{bmatrix} \boldsymbol{\varphi}_{pos} \\ \boldsymbol{\varphi}_{ang} \end{bmatrix}}_{\boldsymbol{\varphi}} - \boldsymbol{\delta}_{pos}^k \end{aligned} \tag{3.58}$$

where $\boldsymbol{\varphi}_{pos} = [\varphi_1, \varphi_2]^\top = [l_x, l_y]^\top$ and $\boldsymbol{\varphi}_{ang} = [\varphi_3, \varphi_4]^\top = [\cos l_\theta, \sin l_\theta]^\top$. Over each path segments the value of $\boldsymbol{\delta}_{pos}^k$ is computed according to Equation (3.4). The formula holds for both Ackermann and Dual Drive models using the proper values of curvature radius r_κ^k. The vector $\boldsymbol{\varphi}_{ang}$ is subject to constraint $\boldsymbol{\varphi}_{ang}^\top \boldsymbol{\varphi}_{ang} = 1$ to satisfy trigonometric consistency that can be written as

$$h(\boldsymbol{\varphi}) = \boldsymbol{\varphi}^\top \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & \mathbf{I}_2 \end{bmatrix}}_{\mathbf{W}} \boldsymbol{\varphi} - 1 \tag{3.59}$$

The error function can be chosen in order both to properly represent a distance from the consistent estimation and to allow the computation of its minimum. Such function must depend on all the measurements collected by the robot, while moving along the n path segments. Although more complex functions could weigh the different components of $\mathbf{e}^k_{pos}$, it is convenient to use the square sum function defined as

$$\begin{aligned}
E(\boldsymbol{\varphi}) &= \sum_{k=1}^{n} {\mathbf{e}^k_{pos}}^\top \mathbf{e}^k_{pos} \\
&= \sum_{k=1}^{n} (\boldsymbol{\varphi}^\top \mathbf{Q}_k - {\boldsymbol{\delta}^k}_{pos})^\top (\mathbf{Q}_k\, \boldsymbol{\varphi} - \boldsymbol{\delta}^k_{pos}) \\
&= \boldsymbol{\varphi}^\top \mathbf{M}_\varphi \boldsymbol{\varphi} - 2\boldsymbol{\varphi}^\top P_\varphi + \left(\sum_{k=1}^{n} {\boldsymbol{\delta}^k_{pos}}^\top\, \boldsymbol{\delta}^k_{pos} \right)
\end{aligned} \tag{3.60}$$

where

$$\mathbf{M}_\varphi = \sum_{k=1}^{n} Q_k^\top \, Q_k = \begin{bmatrix} m_1 & 0 & m_2 & -m_3 \\ 0 & m_1 & m_3 & m_2 \\ m_2 & m_3 & m_4 & 0 \\ -m_3 & m_2 & 0 & m_4 \end{bmatrix} \tag{3.61}$$

$$m_1 = \sum_{k=1}^{n} 2(1-\cos\delta_\theta^k) \tag{3.62}$$

$$m_2 = \sum_{k=1}^{n} \left(\zeta_x^k \,(1-\cos\delta_\theta^k) - \zeta_y^k \,\sin\delta_\theta^k\right) \tag{3.63}$$

$$m_3 = \sum_{k=1}^{n} \left(\zeta_x^k \,\sin\delta_\theta^k + \zeta_y^k \,(1-\cos\delta_\theta^k)\right) \tag{3.64}$$

$$m_4 = \sum_{k=1}^{n} \left((\zeta_x^k)^2 + (\zeta_y^k)^2\right) \tag{3.65}$$

$$\begin{aligned} \mathbf{P}_\varphi &= \sum_{k=1}^{n} \mathbf{Q}_k^\top \, \boldsymbol{\delta}_{pos}^k = \sum_{k=1}^{n} \begin{bmatrix} (\mathbf{I}_2 - \mathbf{R}^\top(\delta_\theta^k)) \, \delta_{pos}^k \\ \mathbf{R}^\top(\zeta_{pos}^k) \, \boldsymbol{\delta}_{pos}^k \end{bmatrix} \\ &= \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \end{bmatrix}^\top \end{aligned} \tag{3.66}$$

Thus, the extrinsic calibration problem is formulated as follows.

Problem 3 (Extrinsic calibration)**.** *Find the parameter vector* $\boldsymbol{\varphi} \in \mathbb{R}^4$ *that*

$$\begin{aligned} \text{minimize} \quad & E(\boldsymbol{\varphi}) = \boldsymbol{\varphi}^\top \mathbf{M}_\varphi \boldsymbol{\varphi} - 2\boldsymbol{\varphi}^\top \mathbf{P}_\varphi + \text{const} \\ \text{s.t.} \quad & h(\boldsymbol{\varphi}) = \boldsymbol{\varphi}^\top \mathbf{W}_\varphi \boldsymbol{\varphi} - 1 = 0 \end{aligned} \tag{3.67}$$

Problem 3 can be solved through method of Lagrange multiplier, since the KKT conditions hold thanks to Slater's conditions. Moreover, it has a closed-form solution that can be obtained by explicitly expanding the linear system $(\mathbf{M}_\varphi - \lambda \mathbf{W})\boldsymbol{\varphi} = \mathbf{P}$ and by substituting the resulting $\boldsymbol{\varphi}$ (as function of multiplier λ) in the constraint $h(\varphi)$. The polynomial equation derived by such substitution is

$$\lambda^2 + b_\varphi \lambda + c_\varphi = 0 \tag{3.68}$$

where the coefficients of the equation are

$$\begin{aligned} b_\varphi &= 2\mu_2 \\ c_\varphi &= \mu_2^2 - \frac{(m_1 p_3 - m_2 p_1 - m_3 p_2)^2 + (m_1 p_4 + m_3 p_1 - m_2 p_2)^2}{m_1^2} \end{aligned}$$

Each $\lambda_{1,2}$ satisfying Equation (3.68) can be back-substituted into the linear system and the two respective solutions $\boldsymbol{\varphi}^{(1,2)}$ can be obtained. The existence of two solutions is due to the symmetries of the reference equations. The additional physical constraint $\varphi_1 \geqslant 0$ is required to choose between the two outputs [18].

3.5 From Theory to Practice

The calibration algorithms for Ackermann and Dual Drive kinematics presented in Sections 3.3 and 3.4 have been implemented and used for calibration of industrial AGVs. The control architecture of the two AGVs has been implemented on industrial PLCs on an embedded computer, meeting the *IEC 61131* standard requirements. PLC systems are widespread and standard industrial solutions to guarantee robustness and real-time execution as well as to comply with safety requirements. Calibration procedures require careful synchronization of data acquired from encoders, laser scanner, sensors and devices in order to relate measurements to the correct segment path, and PLC systems facilitate this task. However, there are some drawbacks in porting advanced algorithms from laboratory to industrial engineered systems. For example, available PLC library suites provide limited support for advanced data structures, linear algebra and numeric algorithms. The proposed calibration algorithms and the prior formulation of objective functions have been designed taking into account such limitations. Indeed, adopting closed-form solutions is also recommended to cope with such issues of calibration, beside their theoretical importance.

Furthermore, computationally expensive tasks such as *Moore-Penrose pseudoinverse* evaluation in Section 3.3.1 or eigenvalues computation in Section 3.3.2 are not

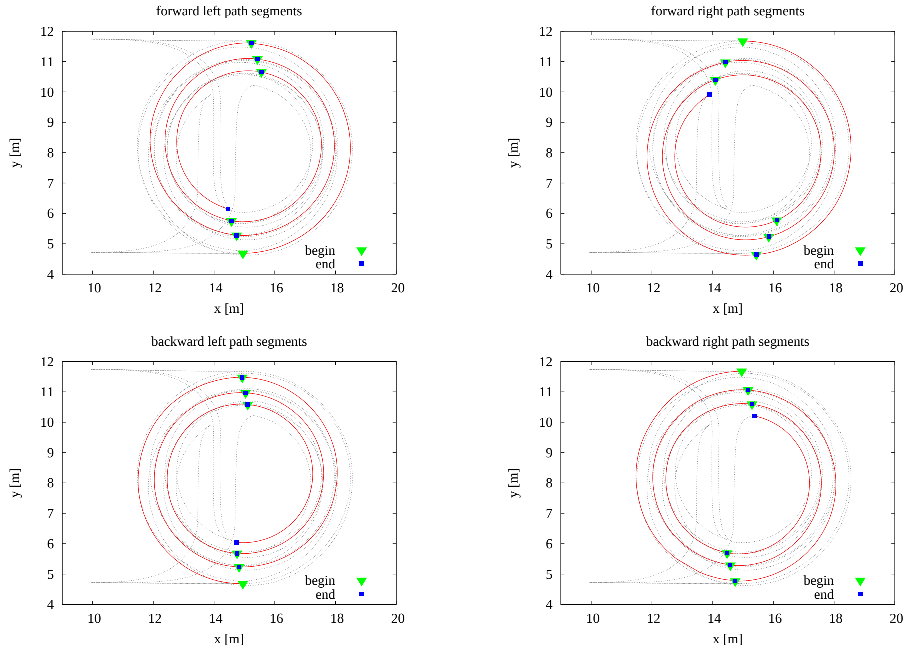


Figure 3.3: An example of path segments used for calibration with $n_{seg} = 6$ grouped according to versus of direction and steering: forward left segments (top-left), forward right segments (top-right), backward left segments (bottom-left) and backward right segments (bottom-right).

an issue for real-time execution on PLCs; despite the AGV localization algorithm, the computation of calibration parameters is not involved any in real-time deadline, so its execution can occasionally be split over multiple PLC cycles without affecting vehicle functioning. Nevertheless, given the reasonable sizes of the matrices involved in the process, calibration calculations are not computationally critical in most cases.

The proposed procedure requires the acquisition of sensor and wheel data while the AGV performs segments of circular paths. For each absolute value of steering angle the system collects four path segments corresponding to left and right steering trajectories, which are travelled in both forward and backward directions. If n_{seg} absolute steering angle values are chosen, then calibration is performed using the corresponding $4n_{seg}$ values. The steering angles of left and right wheels are computed by setting the steering angle of a virtual wheel positioned between the two front wheels. The steering angle α_{wv} of such virtual wheel is varied in the interval $[\alpha_{wv,min}, \alpha_{wv,max}]$. The steering angles α_{wl} and α_{wr} are related to α_{wv} by

$$\tan\alpha_{wl} = \frac{\tan\alpha_{wv}}{1 - \frac{f_y}{f_x}\tan\alpha_{wv}} \qquad \tan\alpha_{wr} = \frac{\tan\alpha_{wv}}{1 + \frac{f_y}{f_x}\tan\alpha_{wv}} \tag{3.69}$$

The minimum and maximum values of α_{wv} are picked in order to keep the robot in the designated calibration area and to avoid physically impossible values of left and right steering angles. The quadruple of path segment i corresponds to

$$\alpha_{wv}^{k} = \alpha_{wv,min} + \frac{\alpha_{wv,max} - \alpha_{wv,min}}{n_{seg} - 1}\, i \tag{3.70}$$

where $i = 0, \ldots, n_{seg} - 1$. Positive α_{wv}^{i} lead to left steering circular path segments whereas $-\alpha_{wv}^{i}$ represent the specular right steering segments. While Equations (3.69) hold for both Ackermann and Dual Drive, the minimum and maximum steering values depend on the physical dimensions of the specific robot and on the dimensions of the calibration area.

Thus, the calibration is structured in four phases distinguished by the different combinations of motion direction (forward or backward) and steer angle (towards left or right). During each of these phases the AGV performs circular path segments and gradually increases the (absolute) value of the steering angle α_{wv}, while the curvature radius r_{κ}^{i} decreases. To maintain a regular pattern the path segments are circular arcs with fixed arc length π. Since the robot is not calibrated, the resulting arc lengths only approximate such value. This choice has proven to be the best trade-off to allow sufficiently long trajectories and, at the same time, to reduce the duration of the procedure. In this way the robot approximately covers half-circles while performing the path segments. The velocity for the calibration procedure has been set to 1500 $cdeg/s$, a value that represents the angular velocity of the AGV rigid body around the instant center of rotation. Figure 3.3 shows an example of path segments used for calibration with $n_{seg} = 6$ subdivided into four phases according to direction (forward or backward) and steering versus (left or right).

During calibration the correct values of calibration parameters are known only up to their nominal values. Thus, the real trajectories performed for calibrating the robot can be potentially inaccurate. In the developed AGV calibration procedure, the initial nominal values of calibration parameters have been determined as follows. The pose of the laser scanner mounted on the AGV is given by CAD design (usually $l_y = 0$ and $l_\theta = 0$ are assumed) neglecting assembly errors. Driving scales s_{wl} and s_{wr} are assessed from radii and encoder resolution of actuated wheels. Steering offsets α_{loff} and α_{roff} are either assumed to be zero or are set so that the two wheels are approximately parallel and straight. The latter choice limits wheel slipping while the AGV travels along path segments and, thus, enables wheel rotation measurements n_{wl}^k and n_{wr}^k to be more consistent with the real travelled distance.

3.6 Experiments

The proposed calibration algorithms have been tested on industrial AGVs in warehouse buildings. An Ackermann AGV 17 and a Dual Drive AGV 12, both manufactured by Elettric80 S.p.A., were used in the first two experiments reported henceafter. The proposed calibration method requires forward and backward AGV motions along circular path segments as discussed in Section 3.5. Hence, calibration has been executed in an obstacle-free area able to accommodate circular trajectories with maximum radius of 5 m. These experiments have been designed to assess the correctness and the precision of the proposed calibration method. The goal of the first experiment is the evaluation of the calibration parameters at different calibration conditions. The second experiment estimates the positioning precision of the AGVs at operation points. The third experiment has assessed the position precision and consistency of a fleets of Dual Drive and Ackermann AGVs.

n_{seg}	α_{wv}-min/max [deg]	valid of 5	α_{loff} [cdeg]		α_{roff} [cdeg]		s_{wl} [mm/tick]		s_{wr} [mm/tick]		l_x [mm]		l_y [mm]		l_θ [cdeg]	
			avg	std	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
6	$26.41 \div 36.41$	5	-207.99	0.21	-291.80	0.40	0.25008	0.00003	0.25013	0.00003	1652.97	0.22	-8.38	0.41	-42.25	0.44
4	$26.41 \div 41.41$	5	-207.98	0.72	-292.62	0.77	0.24982	0.00004	0.24986	0.00005	1652.07	0.21	-6.04	0.92	-39.88	0.87
6	$26.41 \div 41.41$	4	-209.12	1.22	-292.91	1.05	0.25036	0.00016	0.25035	0.00015	1652.74	0.23	-5.07	1.32	-41.71	0.86
6	$26.41 \div 46.41$	4	-208.31	0.43	-292.08	0.51	0.24969	0.00001	0.24972	0.00003	1652.42	0.34	-5.45	0.46	-41.46	0.44

Table 3.1: Calibration parameters of Ackermann AGV.

n_{seg}	α_{wv}-min/max [deg]	valid of 5	α_{loff} [cdeg]		α_{roff} [cdeg]		s_{wl} [mm/tick]		s_{wr} [mm/tick]		l_x [mm]		l_y [mm]		l_θ [cdeg]	
			avg	std	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
4	$26.00 \div 36.00$	4	-286.20	0.58	-296.28	0.12	0.24410	0.00023	0.24489	0.00026	560.30	1.96	4.99	2.35	59.08	3.34
6	$26.00 \div 36.00$	4	-286.75	0.46	-295.16	0.37	0.24375	0.00009	0.24453	0.00010	559.82	0.96	4.96	0.92	61.93	1.59
4	$26.00 \div 41.00$	5	-293.82	0.80	-293.86	0.55	0.24363	0.00007	0.24443	0.00007	558.98	0.28	4.69	2.07	61.42	0.46
6	$26.00 \div 41.00$	5	-294.04	1.22	-292.30	0.82	0.24348	0.00006	0.24429	0.00006	559.55	0.74	4.93	1.74	60.36	1.53
4	$26.00 \div 46.00$	5	-296.64	3.00	-291.30	1.44	0.24364	0.00007	0.24448	0.00011	558.66	0.24	4.60	3.20	60.86	0.37
6	$26.00 \div 46.00$	5	-299.76	1.18	-290.60	0.97	0.24406	0.00016	0.24487	0.00017	558.64	0.11	5.05	1.77	61.27	0.91

Table 3.2: Calibration parameters of Dual Drive AGV.

3.6.1 Calibration Stability

Due to the unavailability of reliable ground-truth parameters, the stability and robustness of the proposed method are better assessed by repeating the calibration procedure in different settings. Each setting is distinguished by the number of path segments n_{seg} used in the four phases, and by the minimum and maximum values of steering angle α_{wv}. A total of 5 trials have been performed for each setting in order to collect significant statistics. The complete calibration procedure takes about $10-15$ minutes per trial depending on the setting parameters.

Table 3.1 collects the results achieved in 20 calibrations for Ackermann AGV 17. Most of the trials have been executed using $n_{seg} = 6$ path segments per motion phase, while collecting also the results of shorter calibration with $n_{seg} = 4$. An offset of 0.41 deg has been added to all the steering angles α_{wv}^i to satisfy the geometric constraints (available free area, physical limits on steering angles of the AGV, etc.) and to achieve conditions more similar to the tests on Dual Drive AGV 12 illustrated in the following. Two outlier estimations have been observed for settings $n_{seg} = 6$ / $\alpha = (26.41 \div 41.41)$ deg and $n_{seg} = 6$ / $\alpha = (26.41 \div 46.41)$. The outlier has been caused by failed sensor registration over one path segment, likely due to insufficient number of detected reflectors. Outlier measurements can be straight-

		α_{loff} [cdeg]	α_{roff} [cdeg]	s_{wl} [mm/tick]	s_{wr} [mm/tick]	l_x [mm]	l_y [mm]	l_θ [cdeg]
Ackermann AGV	Manual	-200.00	-292.00	0.25483	0.25483	1657.00	0.00	-20.00
	Auto1	-208.43	-292.17	0.25016	0.25017	1652.99	-5.42	-41.02
	Auto2	-208.20	-292.38	0.24968	0.24975	1652.00	-5.94	-41.96
Dual Drive AGV	Manual	-220.00	-298.00	0.24031	0.24031	575.00	12.00	62.00
	Auto1	-292.24	-291.31	0.24350	0.24435	560.24	4.05	58.36
	Auto2	-301.72	-292.24	0.24418	0.24496	558.74	6.68	60.74

Table 3.3: Calibration parameters used for experiments

forwardly detected by monitoring registration conditions (e.g. checking the number of detected and associated landmarks) and easily dealt with in practical application of the method. The mean value and standard deviation of calibration parameters in Tables 1 and 2 have been computed only from the valid calibration trials. The average values of estimated parameters do not significantly change with the different settings. The sensor coordinate l_y is likely the parameter most sensitive to experimental conditions, as shown by its average values and standard deviation, which is slightly higher than l_x. However, the average value of l_y oscillates at most by 3 mm in different conditions. All the variations on angular parameters are less than 1 deg.

Table 3.2 illustrates the results achieved in 30 calibrations for Dual Drive AGV 12. Also for Dual Drive two outlier estimations have been observed in sets $n_{seg} = 4$ / $\alpha = (26 \div 36)$ deg and $n_{seg} = 6$ / $\alpha = (26 \div 36)$ deg. Statistics in Table 3.2 are computed with the exclusion of outlier parameters.

The position parameter most sensitive to experimental conditions is again l_y, as shown by its standard deviation. Parameter l_θ, that must be very well calibrated to achieve good AGV movement, has a steady average and a small standard deviation. The values of estimated parameters are stable to different calibration conditions like the number of segments used for calibration or the interval of steering angles. Outliers seem to affect more the estimated value of parameter l_y, whereas the computed values of other parameters are generally comparable to the estimation without outliers. Standard deviations decrease with higher number of segments or with larger steering angle interval, although there are some exceptions. Using 6 paths instead

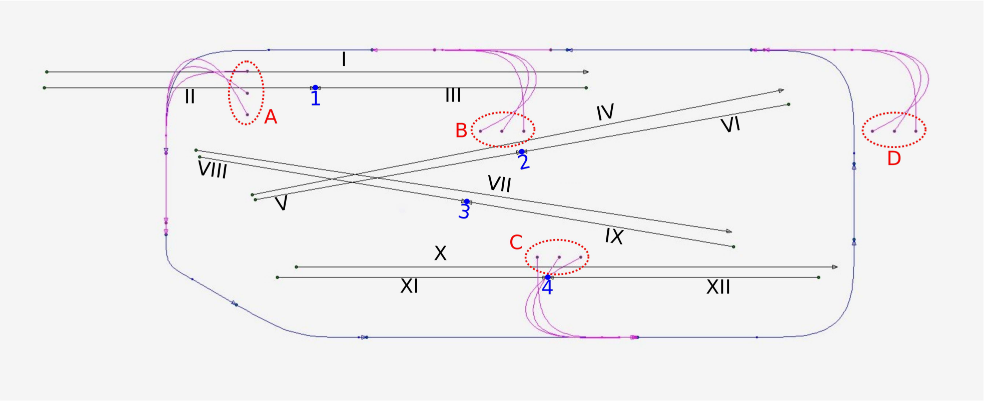


Figure 3.4: The layout of paths used to perform position precision experiments. The operative area for the vehicle is approx. 16.5 m x 5.5 m

of 4 seems to return smaller standard deviations in estimated values. The best set of calibration parameters appears to be the one returned from the tests with $n_{seg} = 6$ / $\alpha = (26 \div 41)$ *deg*, considering the lack of outliers and the high stability of parameters.

3.6.2 Position Precision of Single AGV

Several tasks performed by AGVs require to stop at given operation points of the warehouse, e.g. to load or unload pallets with the forklift. The intrinsic parameters are part of the robot kinematic model and influence both the control system and the odometry. The extrinsic parameters define the pose of the laser scanner used in navigation and localization. An important aim of calibration is to make all the AGVs working in a warehouse stop on the same operation points with adequate precision. Three different sets of experiments have been executed in a real warehouse to evaluate positioning precision obtained by automatic calibration.

1. *Localizer Error*. The first test assesses the distance and heading errors mea-

sured by the localizer, when the AGV stops at given operating points. Distance error represent the lateral position error w.r.t. the AGV trajectory terminating on the operating point. These errors depend on AGV navigation and control system and on accuracy of calibration parameters, and cannot be used as ground-truth. However, smaller absolute errors reveal better calibration parameter. This test has been performed stopping at operation points placed after either straight or uneven high curvature paths that stress the control system. Straight and curved paths are labelled respectively by roman numerals (I, IV, VII and X) and letters (from A to D) in Figure 3.4.

2. *Straightness Error*. The second test assesses the accuracy of sensor orientation l_θ and, indirectly, of all the other calibration parameters. The AGV moves back and forth over straight paths 10 m long (straight lines with roman numerals I, IV, VII and X in Figure 3.4). A perfectly calibrated vehicle should execute perfect and overlapping straight trajectories while moving forward and backward. Otherwise, the AGV produces a ”leaf shape” moving on slightly different paths when moving forward or backward. The larger is the error on l_θ, the larger is the distance between forward and backward paths. The distance is measured approximately on the medium point of the path by marking the AGV position (the side of its chassis) on the ground as shown in Figure 3.5.

3. *Halting Point Precision*. The third test evaluates the global quality of calibration parameter set by measuring the halting point precision. The vehicle moves back and forth on two facing straight paths, each 5 m long. In Figure 3.4, the halting points are labelled with arabic numerals (from 1 to 4), whereas forward and backward paths are in roman numerals. The AGV is manually stopped in the halting point and two points on both sides of back wheels are marked on the ground. This procedure is repeated for both the front facing paths. For example, point 1 is reached from both paths II and III. The halting position of right back wheel from path II is compared with the position of left back wheel on path III (side A), and viceversa (side B). Figure 3.6 shows an example of marked halting points on the side of an AGV rear wheel.

The three tests have been performed for both Ackermann AGV 17 and Dual Drive AGV 12. For both vehicles, we compared the results obtained with calibration parameter sets *Manual*, *Auto1* and *Auto2* (the latter is not used in some tests) shown in Table 3.3. As suggested by their name, set *Manual* is obtained through manual calibration performed by an expert operator, whereas *Auto1* and *Auto2* are computed in two different calibration trials of the proposed method for each vehicle type.

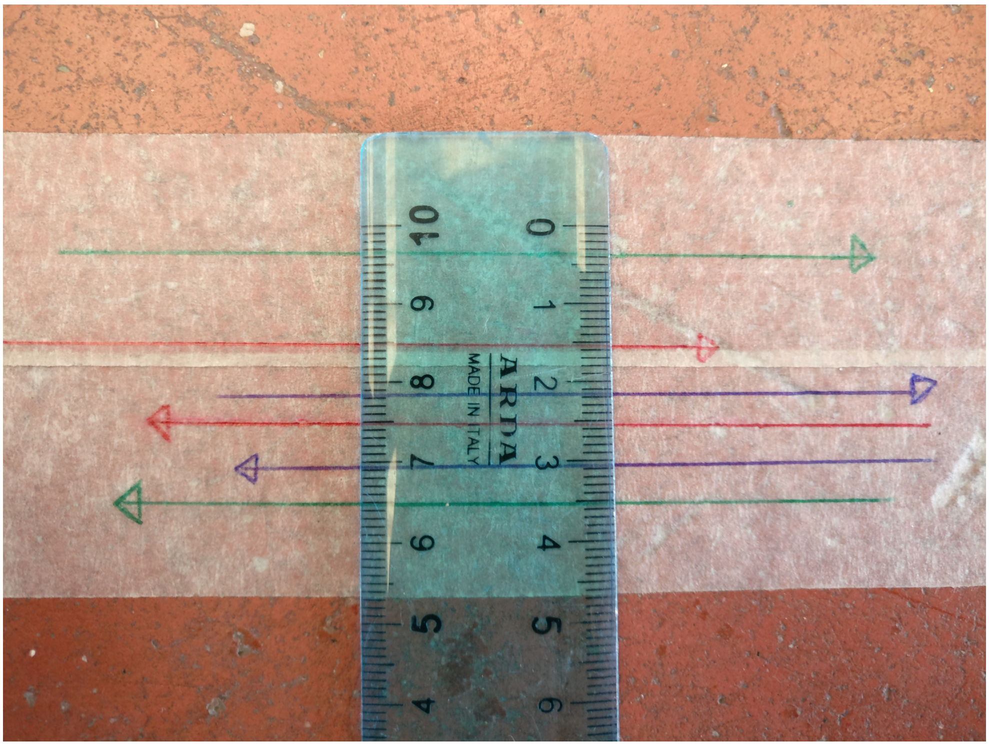


Figure 3.5: Example of forward and backward paths marked on the ground during the execution of *Straightness Error* test on Ackermann AGV 17. The marker line color identifies the paths obtained using calibration parameters *Manual* (green), *Auto1* (blue) and *Auto2* (red).

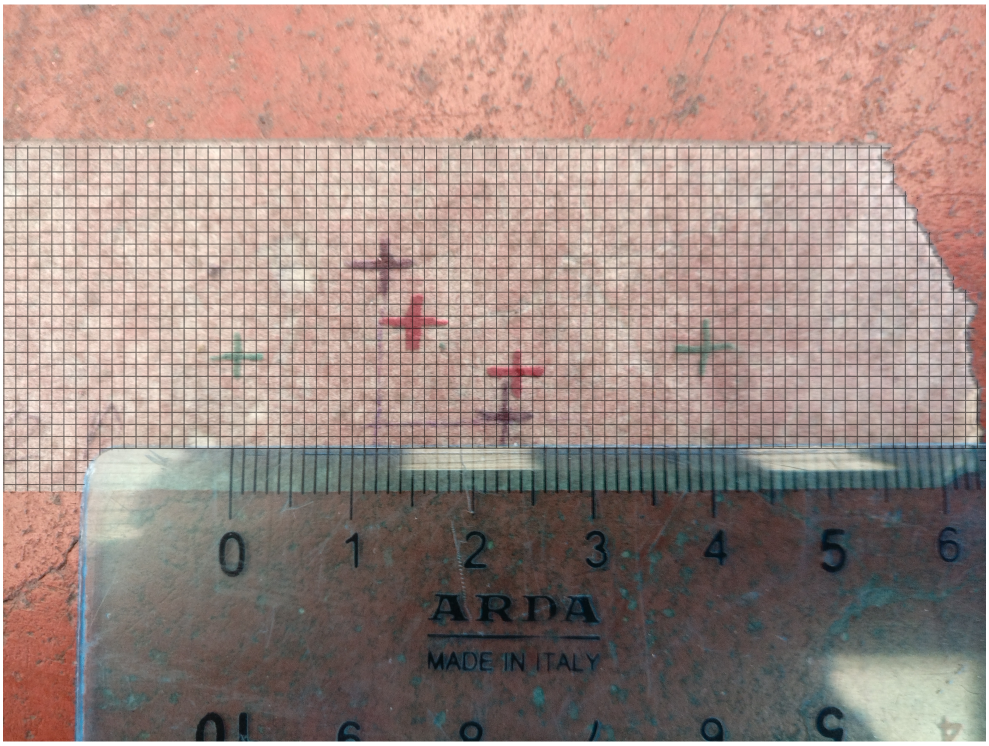


Figure 3.6: Example of halting points marked on the side of AGV rear wheel during the execution of *Halting Point Precision* test on Dual Drive AGV 12. The marked point color identifies the points obtained using calibration parameters *Manual* (green), *Auto1* (blue) and *Auto2* (red).

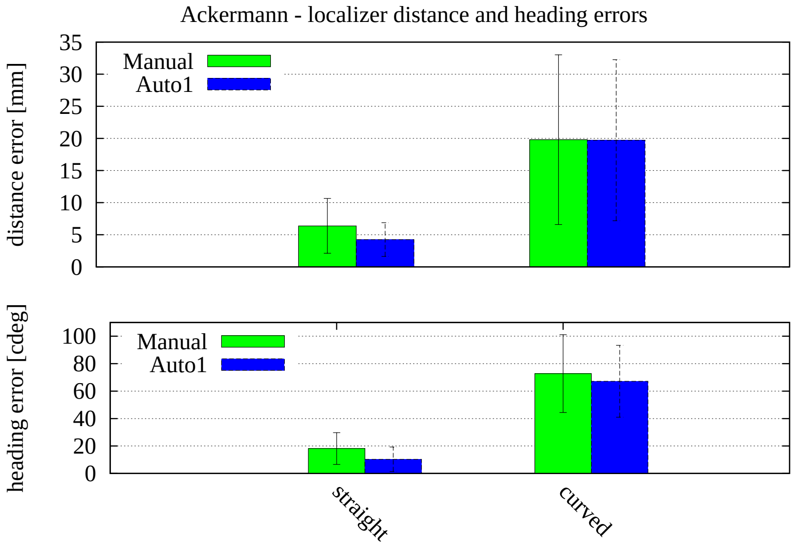


Figure 3.7: Results of *Localizer Error* test for Ackerman AGV 17 on four straight and four curved path segments.

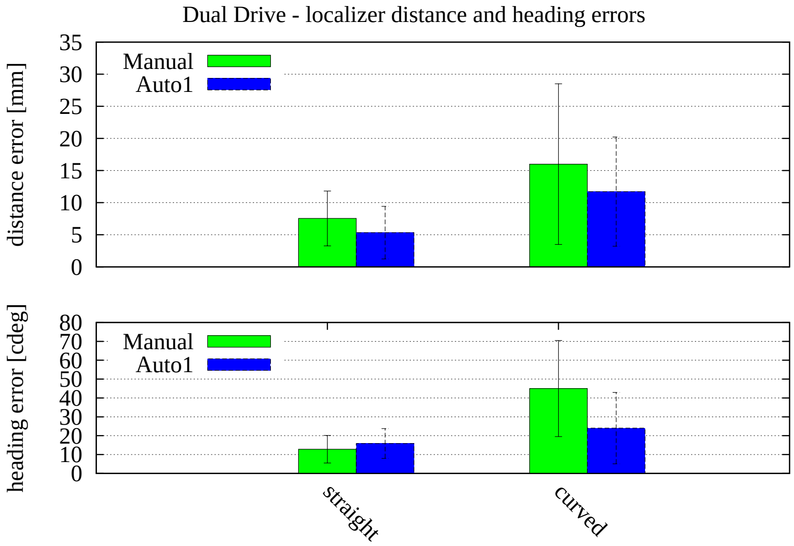
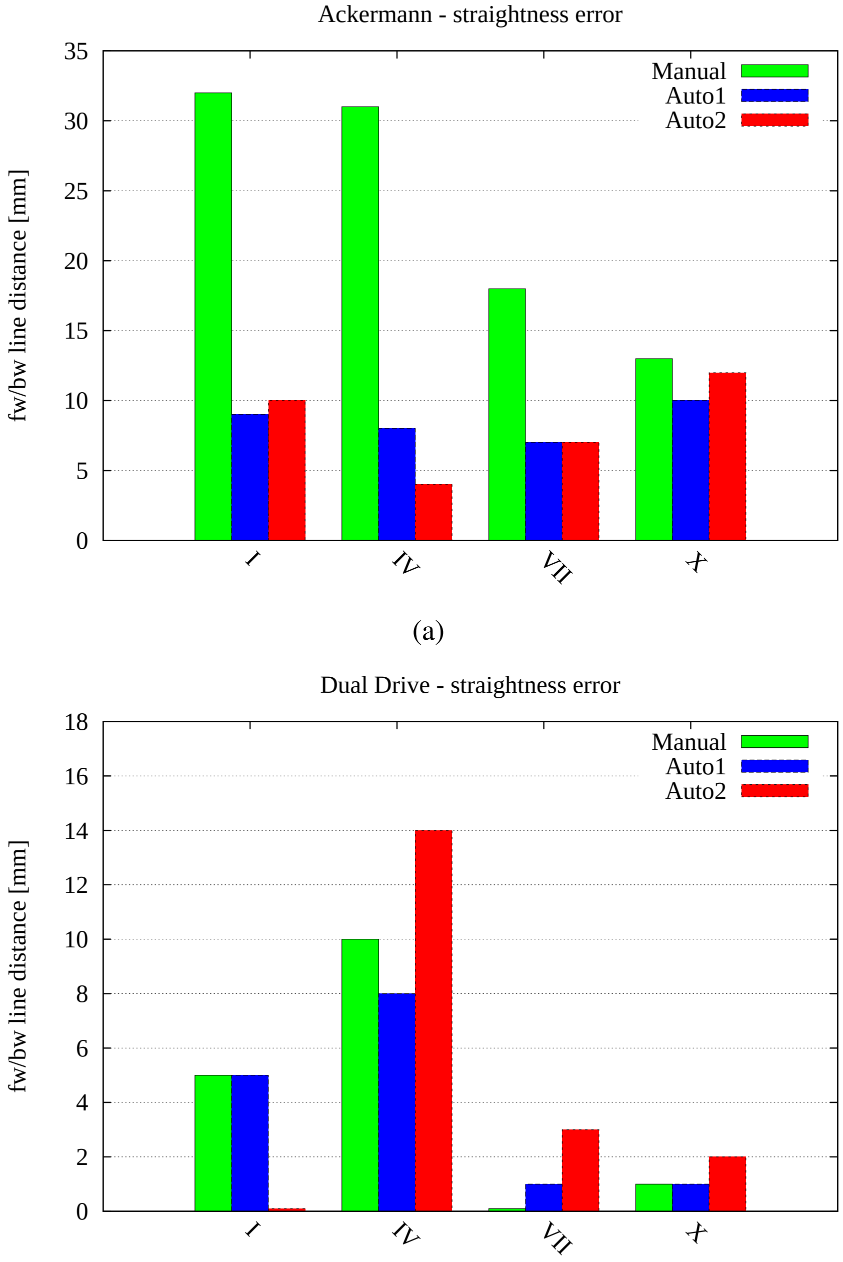


Figure 3.8: Results of *Localizer Error* test for Dual Drive AGV 12 on four straight and four curved path segments.

The statistical results of the *Localizer Error* test are displayed by the histograms of distance and heading errors in Figures 3.7 and 3.8 for respectively Ackermann and Dual Drive. Automatic calibration achieves lower average values and standard deviations of distance and heading errors than manual calibration for the two models. The error due to trajectory tracking control is more pronounced on curved paths, but automatic calibration performance is better or comparable.

The *Straightness Error* test has assessed the accuracy of parameter l_θ and, indirectly, of the other calibration parameters. Figure 3.9(a)-(b) shows the histogram of errors obtained on four straight paths. On Ackermann AGV 17 the automatic calibration procedure always outperforms the manual one. For Dual Drive AGV 12, the straightness error of the calibration parameter sets *Auto1* and *Auto2* is comparable with the one achieved by manual calibration. All trials demonstrate that the value of l_θ has been correctly estimated by manual as well as automatic calibration.

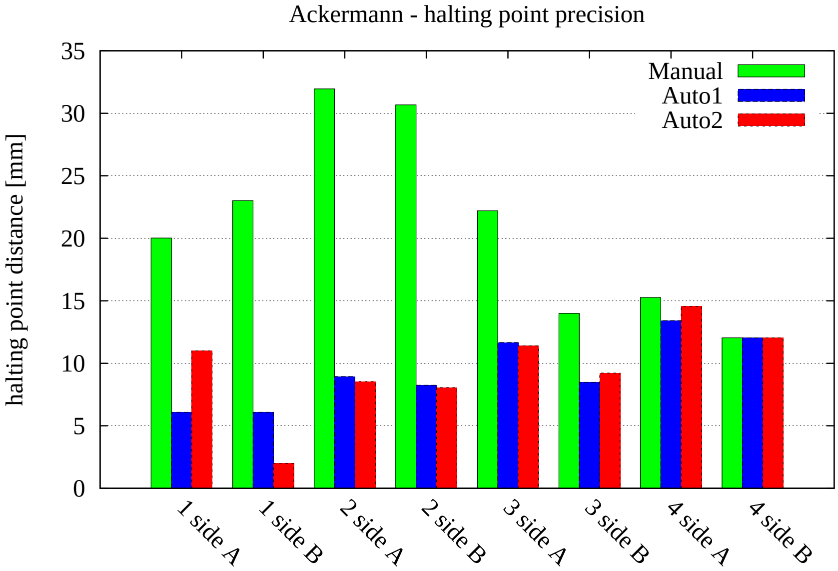
In the *Halting Point Precision* test, we measured the precision on halting points reached from two different paths. The distance between the points obtained by the AGV moving on the two facing paths are summarized in Figure 3.10(a)-(b). For both Ackermann AGV 17 and Dual Drive AGV 12, the automatic calibration procedure achieves lower (or, in the worst cases, similar) distance value between the trail points than the manual one. Parameter computed with the automatic calibration appear to yield more stable and accurate halting points.



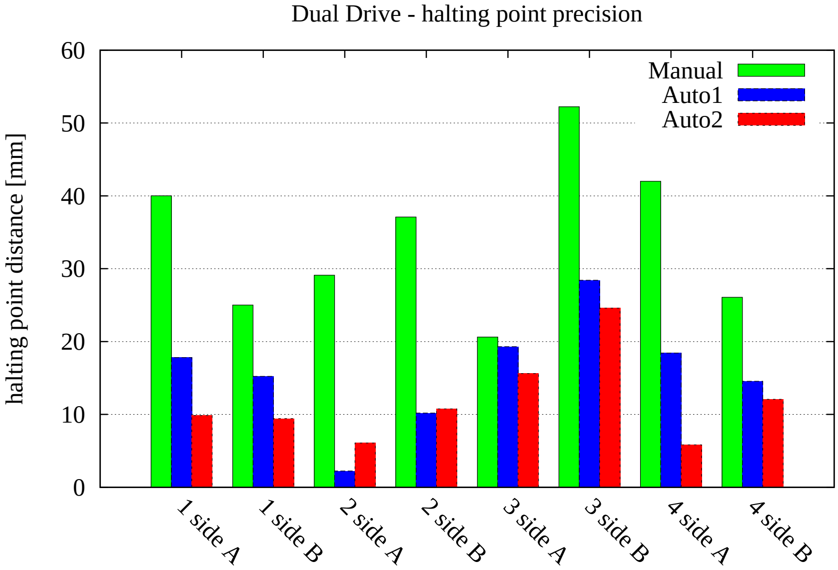
(a)

(b)

Figure 3.9: Results of *Straightness Error* test for Ackerman AGV 17 (a) and Dual Drive AGV 12 (b) on four straight segments labelled by roman numerals as in Figure 3.4.



(a)



(b)

Figure 3.10: Results of *Halting Point Precision* test for Ackermann AGV 17 (a) and Dual Drive AGV 12 (b) on four halting points 1, 2, 3 and 4 in Figure 3.4.

3.6.3 Position Precision of AGV fleets

The results presented in the previous Section estimate the position precision achieved by a single AGV when calibrated manually or using the proposed method. Several tasks performed by AGVs require to stop at given operation points of the warehouse, e.g. to load or unload pallets. Position precision depends on different factors including the localization system and the accuracy of calibration. Thus, an important goal of calibration is to make all the AGVs halt at the same points with high precision.

In this Section, the position precision of fleets of Dual Drive and Ackermann AGVs operating in a real industrial warehouse is reported. This part of experiments has been impossible to test on Ackermann fleet due to AGV unavailability at the moment of the trials.

The tests on *straightness error* and *halting point precision* discussed in Section 3.6.2 have been performed on a set of five Dual Drive AGVs calibrated with the proposed method. Table 3.4 shows the obtained calibration parameters used for each AGV. It can be observed that, although the AGVs are built according to the same design, there are significant variations in both intrinsic and extrinsic parameters and more noticeably in the latter ones. In particular, it is difficult to mechanically mount the laser scanners with the same orientation l_θ.

Figure 3.11 reports the straightness error of the five AGVs measured for two different straight paths F_I and F_{II}. As usual the straightness error is taken in the middle of the straight path when travelled forward and backward. The measured distance is less than 8 *mm* for all the AGVs.

$AGVn^{\circ}$	α_{loff} [cdeg]	α_{roff} [cdeg]	s_{wl} [mm/tick]	s_{wr} [mm/tick]	l_x [mm]	l_y [mm]	l_{θ} [cdeg]
1	-325	-286	0.23847	0.23783	554	-8	-18
2	-350	-261	0.23690	0.23720	560	-11	-139
3	-299	-209	0.23703	0.23714	549	-4	-6
4	-369	-244	0.23639	0.23641	548	-2	-44
5	-388	-247	0.23590	0.23620	554	-11	59

Table 3.4: Calibration parameters of the five Dual Drive AGVs used in fleet position experiments.

$AGVn^{\circ}$	α_{loff} [cdeg]	α_{roff} [cdeg]	s_{wl} [mm/tick]	s_{wr} [mm/tick]	l_x [mm]	l_y [mm]	l_{θ} [cdeg]
1	-281	-285	0.25347	0.25355	1175	-7	-109
2	-246	-265	0.25373	0.25367	1763	-7	49
3	-258	-283	0.25314	0.25324	1763	1	46
4	-275	-289	0.25374	0.25368	1765	-6	8
5	-277	-282	0.25385	0.25369	1764	-5	71
6	-307	-308	0.25310	0.25314	1774	-14	-73
7	-294	-285	0.25341	0.25334	1752	-9	-1
8	-277	-265	0.25309	0.25319	1779	-11	20

Table 3.5: Calibration parameters of the eight Ackermann AGVs used in fleet position experiments.

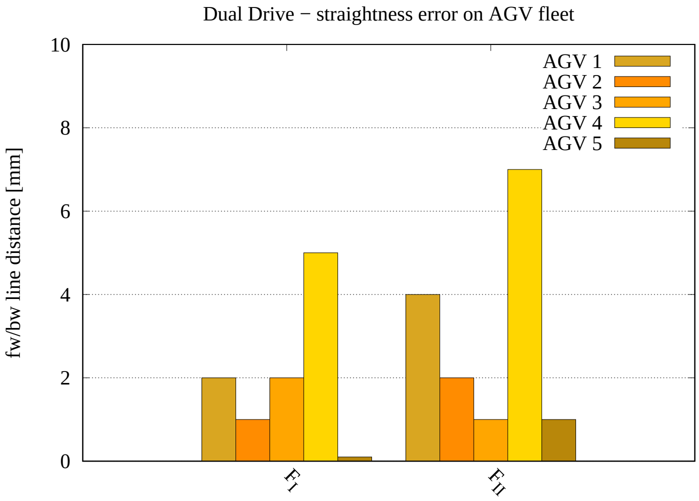


Figure 3.11: Results of *Straightness Error* test for for a fleet of five Dual Drive AGVs on two straight paths F_I and F_{II} in an industrial warehouse.

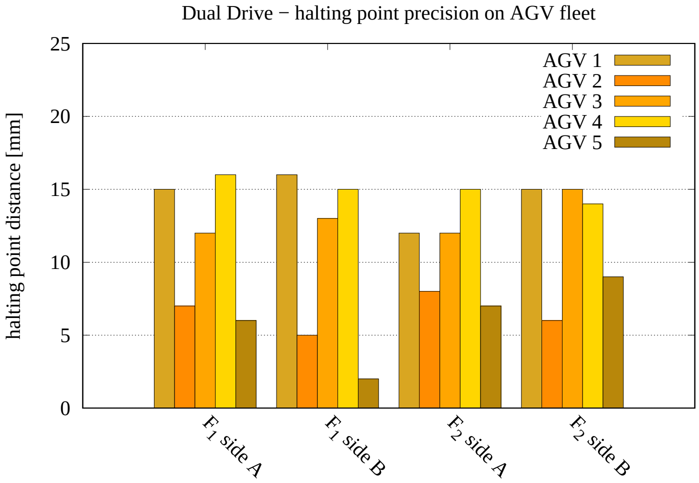


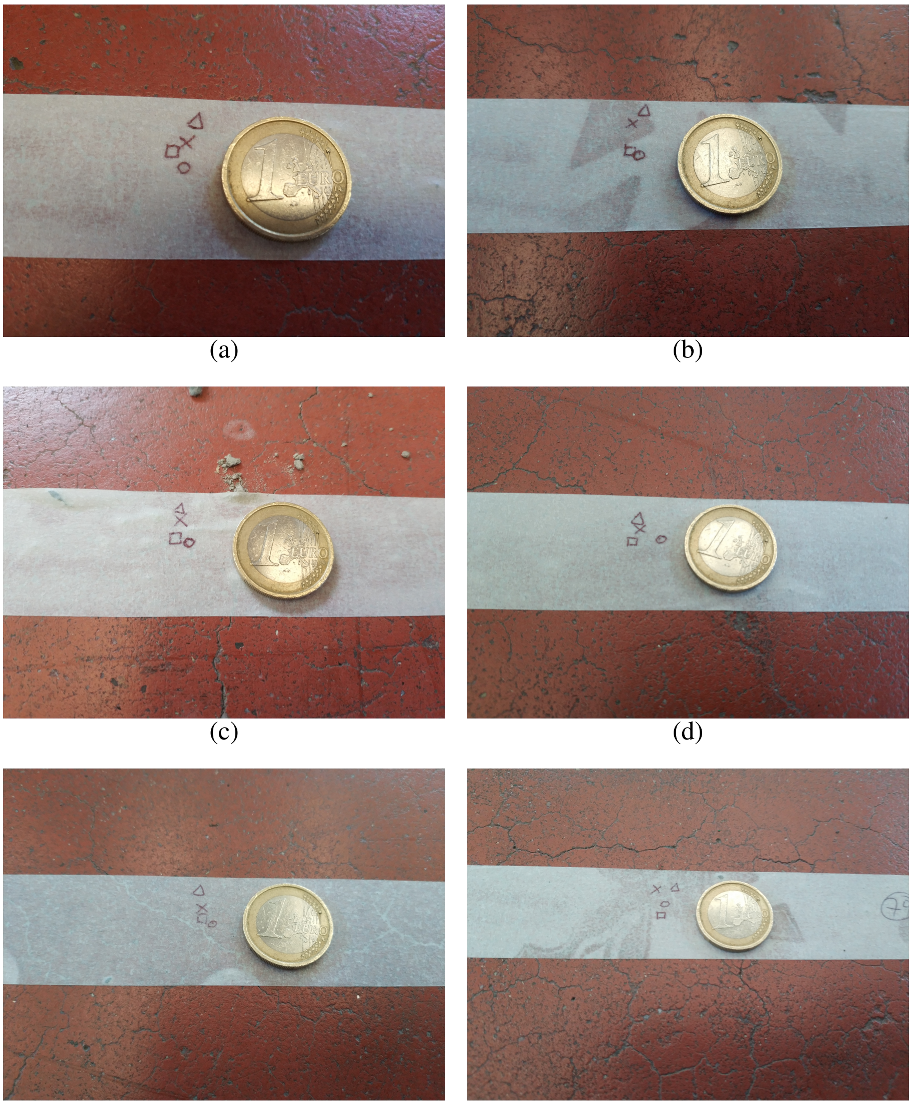
Figure 3.12: Results of *Halting Point Precision* test for a fleet of five Dual Drive AGVs on two halting points F_1 and F_2 in an industrial warehouse.

Tests on halting point precision have been performed by stopping the robots on two points F_1 and F_2. Each point is reached from two opposite directions and the position error is measured on both the front wheels (sides A and B). The maximum wheel displacement is at most 16 *mm*. The resulting positional accuracy is sufficient for correct pallet loading and unloading operations. Indeed, the proposed automatic procedure has consistently calibrated AGVs, enabling them to perform their tasks with equivalent accuracy in a real industrial plant.

Furthermore, a group of experiments on stopping point precision of fleets of Dual Drive and Ackermann AGVs has been performed. Four Dual Drive AGVs out of five of the fleet described above have been moved on different points of a specific industrial plant. The same approach has been followed with the full fleet of eight Ackermann AGVs. For each AGV, a stopping point has been marked on the floor with different symbols over each trial point. The experimental points have been chosen at the end of straight and curved segments, reached in both forward and backward motion of the AGV. Figures 3.13 and 3.14 shows the results for Dual Drive and Ackermann AGV fleets.

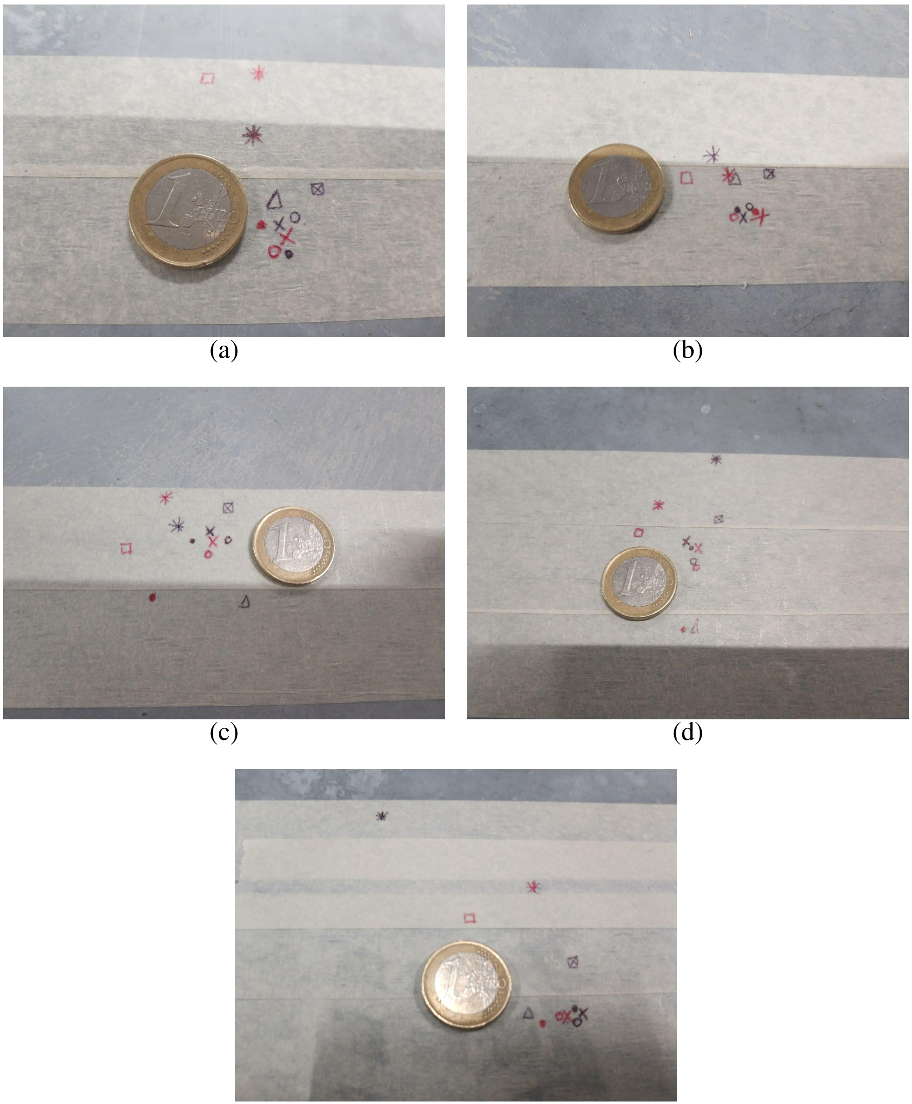
The results of the Dual Drive AGV fleet are displayed in Figure 3.13. The AGV stopping points are marked on the floor with different symbols. The distance between the points is at most 15 *mm*, that is enough to ensure adequate positioning performance.

The results of the Ackermann AGV fleet are displayed in Figure 3.14 instead. The AGV stopping points are marked on the floor with different shape and colors: over the eight automatically calibrated AGVs, three manually calibrated AGVs have been testes. The latter three vehicles are marked using the red square (□), the red asterisk ($*$) and the black asterisk ($*$). The distance between the points of automatically calibrated AGVs is around 20 *mm* while the points of manually calibrated AGVs arise as outlier, mainly in Figures 3.14a, 3.14d and 3.14e. The automatic calibration shows higher positioning precision than the manual one, returning considerably closer stopping points.



(a) (b) (c) (d) (e) (f)

Figure 3.13: Stopping point marked on the floor for a fleet of automatically calibrated Dual Drive AGVs.



(a) (b)

(c) (d)

(e)

Figure 3.14: Stopping point marked on the floor for a fleet of automatically calibrated Ackermann AGVs. **Note:** as a comparison, the red square (□), the red asterisk ($*$) and the black asterisk ($*$) refer to manually calibrated Ackermann AGVs.

3.7 Discussion

This chapter has presented a novel method for extrinsic and intrinsic automatic calibration of four wheel industrial AGVs compliant with Ackermann and Dual Drive kinematics. The algorithm estimates the trajectories measured by an onboard sensor and the expected ones given by the state of the wheels. By means of the model equations derived for the specific kinematics, both intrinsic and extrinsic calibration parameters are computed through closed-form solutions of least-square optimization.

The methods have been implemented on PLC controllers and experiments have been carried out with industrial AGVs in a warehouse. The precision of the estimated AGV parameters in repeated calibration trials is at most 0.1 deg for angular parameters and typically less than 5 mm for position parameters. The accuracy can be improved by a proper selection of the path segment executed by the AGV and by a manual straightening/parallelization of wheels. The AGVs calibrated with the proposed methods have shown the ability to stop at the same operation points with a typical accuracy of $10-15$ mm if the control system is not overstressed.

The experiments on a fleet of AGVs show that all the vehicles calibrated with the proposed method achieve adequate similar position precision. This level of position accuracy is comparable with the accuracy obtained with the best manual calibration available in industrial environment, or even higher as shown in the stopping point precision trial. Furthermore, the proposed methods achieve better or similar results w.r.t. other complete calibration approaches (i.e. tuning intrinsic and extrinsic parameters) tested on mobile robots with different kinematics, such as [18, 19].

With the suggested number of path segments the automatic calibration procedure takes about 15 minutes for each AGV instead of more than one hour required by the manual iterative procedure. The availability of an automatic, fast and reliable calibration procedure allows more frequent recalibration and hence better AGV localization and navigation performance.

Chapter 4

Localization in Unstructured Environments

High precision, pin-point repeatability and solid performance are crucial for robust AGV positioning in industrial environments. Besides the well-established artificial landmark approaches, several novel techniques have been proposed, yet their real-world application has still to be proven.

This chapter presents three different methods for laser-based AGV localization, to assess their performance in real industrial plants. The three approaches correspond to sensor models for the EKF localization used for AGV navigation. The first localization algorithm is based on detection of keypoint feature corresponding to high curvature regions of scans. The registration-based algorithms estimate the position and orientation displacement between range measurements and previously computed dense map. Moreover, the latter two algorithms embody a virtual scan estimation to improve pose evaluation in vast industrial plants.

The aforementioned map of the industrial environment is obtained using a Terrestrial Laser Scanner and innovative software processing, to attain the most accurate ground-truth for localization. Finally, these methods are compared with traditional landmark-based localization using reflectors. Measurement ground-truth is provided by a Total Station with sub-millimetre accuracy.

4.1 Introduction

The AGVs commonly used in industrial warehouse logistics [18, 61] must be able to localize to reach the desired operation point with high accuracy and without collision with people and obstacles. Their navigation is usually based on reliable laser sensing technology to guarantee accurate pose estimation and compliance to strict safety regulations.

The evolution of industrial AGVs (w.r.t. perception, localization, mapping and navigation) is rarely documented by application-oriented works that adapt the mainstream research advancements. Several works focus on the perception of objects (e.g. pallets, products, etc.) and landmarks using different sensors like lasers [62], cameras [63] or RGBD sensors [64]. In [65] the pose of industrial AGVs is estimated using both artificial landmarks and temporary landmarks extracted from measurements in the regions devoid of artificial ones. Some complete industrial navigation systems have been presented. The work in [66] illustrates the navigation system of AGVs intended for partially structured warehouses and with frequent changes in the floor plant layout. Such system integrates the localization based on reflective landmarks with local maps, which enable pallet management and obstacle avoidance. Other works [15] focus on the construction of 3D map for estimation of floorplan and 2D navigation map.

Despite the large literature on localization, these papers rarely refer to real industrial applications meeting safety requirements, robustness and high technology readiness level. Localization accuracy is seldom carefully assessed by comparing different approaches in the same industrial setup. Moreover, the vast majority of the non-canonical approaches aforementioned are poorly integrated in industrial contexts: their robustness and repeatability are often not adequate to industrial requirements. Thus, standard localization is still based on detection of artificial landmarks made of reflective material with cylindrical shape, usually named reflectors. Reflector placement is a time-consuming and critical operation that must guarantee complete coverage and avoid symmetries. The observed reflectors are matched with static landmark maps. More advanced perception and estimation methods can increase the

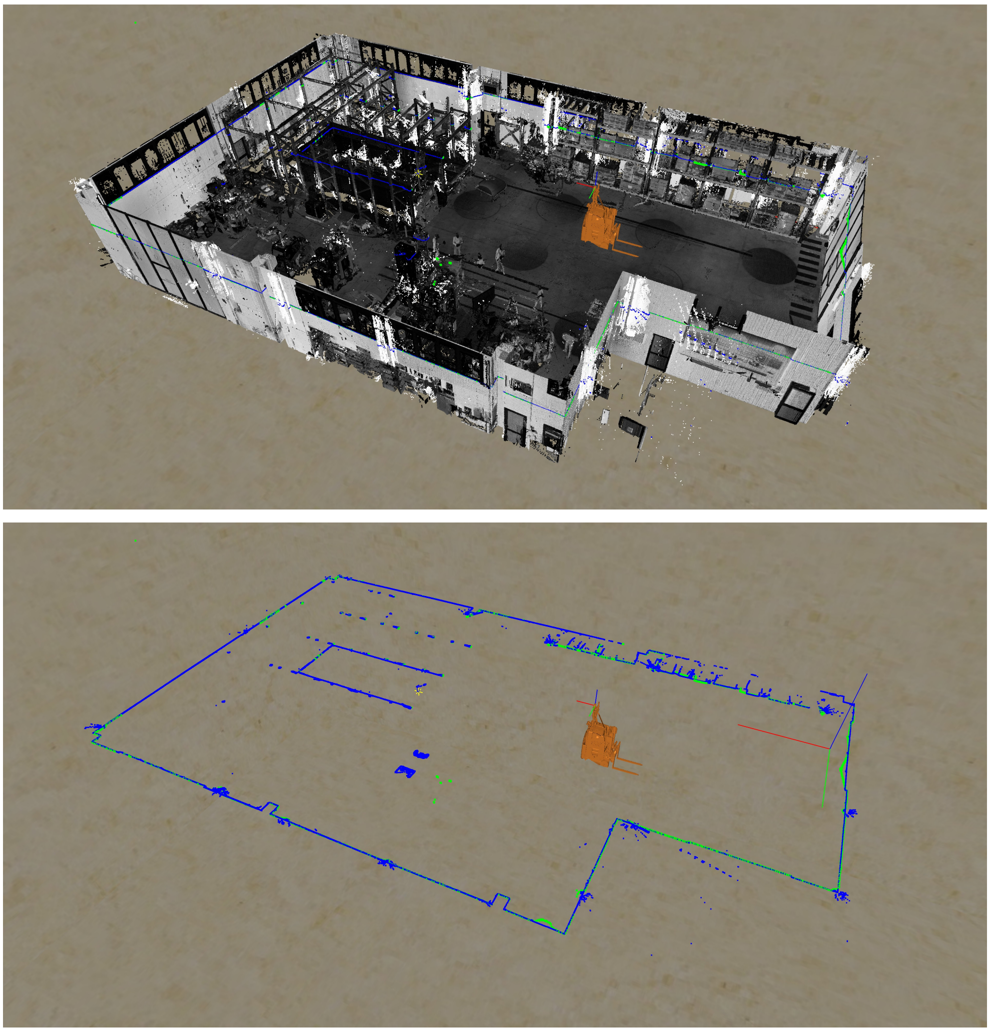


Figure 4.1: Top: 3D TLS survey of an industrial plant (ceiling intentionally removed) with extracted map (blue) and matched laser scan (green) with localized AGV (orange). Bottom: same scenario without the 3D point cloud.

autonomy of industrial AGVs and the flexibility of plant setup.

This chapter presents three localization methods for AGV guidance in industrial plants and their experimental assessment. The localization algorithms match the range finder measurements with a floorplan map according to Extended Kalman Filter (EKF) framework [67]. The floorplan map is computed from the 3D point cloud map obtained by the Terrestrial Laser Scanner (TLS) mapping during the plant setup [68]. In particular, the planar representation is achieved by slicing the 3D point cloud at AGV laser scan height from the carefully extracted floor [69] (see Figure 4.1). Such planar map is used to detect landmarks or to compute a virtual laser scan from a given viewpoint, that is later exploited by the different sensor models.

The first sensor model detects *SKIP* (SKeleton Interest Point) keypoint features corresponding to high curvature points in laser scans and matches these salient points to the map landmarks. The second and the third methods, namely *ICP* and *ARS-Grid*, correct localization estimation through registration of the actual laser scan with the virtual scan. The robust ICP [70] algorithm efficiently and accurately aligns the current range scan with the map, in particular when a reliable initial guess of the rigid transformation is available. The closest point search is sped-up by a grid search data structure overlapped with the floorplan map. *ARS-Grid* decouples the evaluation of orientation-based on *Angular Radon Spectrum* (ARS) [71] and of translation-based on grid correlation. Experiments assess the localization precision of the proposed methods compared with standard industrial localization based on artificial reflectors.

The main contributions of this chapter are the following:

- the estimation of the map used in localization from the 3D model obtained from the TLS survey integrated in the industrial workflow;
- three methods exploiting a novel SKIP feature, an ICP implementation based on efficient closest point search and a novel registration algorithm based on ARS and grid correlation;
- the experimental comparison of the proposed methods with standard reflector localization methods.

The chapter is organized as follows: Section 4.2 presents the EKF localization framework for the proposed methods. Section 4.3 shows the creation of the map, the grid search data structure and the virtual scan approach. Section 4.4 illustrates the proposed algorithms acting as sensor models. Section 4.5 discusses the industrial setup. Section 4.6 presents the experiments performed on real industrial AGVs. Section 4.7 concludes the chapter.

4.2 AGV Localization

We consider an AGV, equipped with laser sensors, moving in a structured environment with known a priori map. With respect to this scenario, the localization problem is that to determine the pose of the vehicle on the base of the laser information and the prior maps of the environment. The localization of industrial AGVs is commonly addressed by standard Extended Kalman filter (EKF). Let $s(k) \in \mathbb{R}^2 \times S$ be the state corresponding to AGV pose in global frame, $u(k)$ the drive signals to the vehicle, $z(k)$ the sensor measurements at time k. The motion model of the system is given by the equations

$$s(k) = g(s(k-1), u(k)) + w(k), \tag{4.1}$$

where $g(\dots)$ is the car-like kinematic model of AGV [18, 61].

The three localization algorithms described in the following Section are based either on keypoint features extracted from the laser scan or on the registration between the current scan and the map. The proper sensor model is the direct estimation of AGV pose represented by

$$h(s) = s^{obs}, \tag{4.2}$$

where s^{obs} is the position of the vehicle observed via registration between a local map and current scan at the current vehicle location s.

Let $H^L = \{h_p^L\}_{i=p,\dots,P}, h_p \in \mathbb{R}^2$ be a set of scan points in the laser frame with the origin of the scan at $0 \in \mathbb{R}^2$. The same set of point in the vehicle frame is called H^V and it is obtained with the invariant time transformation T_L^V:

$$T_L^V h_p^L = \begin{bmatrix} \cos\theta_v & -\sin\theta_v \\ \sin\theta_v & +\cos\theta_v \end{bmatrix} h_p^S + \begin{bmatrix} x_v \\ y_v \end{bmatrix} \tag{4.3}$$

where the vector $[x_v, y_v, \theta_v]$ represents the position of the center of the laser in the vehicle frame.

Now we consider a submap $H^{\hat{V}}$ in the vehicle frame, which is generated through ray tracing on global map from current estimated vehicle pose $s^G = [s_x^G, s_y^G, s_\theta^G]$ in global frame G. In this way we define the pose $\xi^{\hat{V}} = [x_\xi, y_\xi, \theta_\xi]$ of a current scan H^V in the submap frame as the transformation $T_V^{\hat{V}}$:

$$T_V^{\hat{V}} h_p^V = \begin{bmatrix} \cos\theta_\xi & -\sin\theta_\xi \\ \sin\theta_\xi & +\cos\theta_\xi \end{bmatrix} h_p^V + \begin{bmatrix} x_\xi \\ y_\xi \end{bmatrix} \tag{4.4}$$

where $t_\xi = (x_\xi, y_\xi)^T$ describes the translation and θ_ξ the rotation which rigidly transforms the scan points h_p^V from the current vehicle frame into the submap frame.
The new estimated pose of the vehicle in the global frame G can then be found by a composition of transformations $T_{\hat{V}}^G$ and $T_V^{\hat{V}}$.

Since the transformation $T_{\hat{V}}^G$ is given by current estimated pose s^G, the problem of finding the observation s^{obs} relies in finding the pose $\xi^{\hat{V}}$ of the current scan points H^V with respect to the submap frame.

In the next Section, the three different approaches that we use to find an estimate of s^{obs} are presented.

4.3 Map and Virtual Scan

4.3.1 Map Construction

The localization system presented in this chapter relies on the 3D model of the industrial plant obtained from a TLS survey [68, 69]. Figure 4.1 (top) shows an example of the obtained 3D map. The AGVs are provided with navigation laser scanners mounted at known height from the ground. A realistic floorplan map is obtained by slicing the 3D point cloud at laser sensor level w.r.t. the ground, which is segmented as in [69]. The Section is sliced with a certain width (usually a few centimetres) and then the points are projected on laser sensor plane. Refinement steps proceed as explained in [69]. Figure 4.1 (bottom) illustrates the outcome of the procedure. The obtained planar model enables both the automatic extraction of corner landmarks for SKIP feature localization and the dense registration required by ICP and ARS-Grid.

Algorithm 1 Grid Search Definition

```
Inputs: $\mathscr{P} = \{p_k\}_{k=0,\dots,n-1}$ with $p_k = [x_k \; y_k]^T$: map points,
        $r$: grid cell size,
        $x_{max}, x_{min}, y_{max}, y_{min}$: maximum and minimum x and y coordinates of $\mathscr{P}$,
Output: $\mathscr{G}$: 2D grid structure
procedure GRID DEFINITION($\mathscr{P}, r$)
    $s_x = 1 + \left\lceil \frac{x_{max} - x_{min}}{r} \right\rceil$;
    $s_y = 1 + \left\lceil \frac{y_{max} - y_{min}}{r} \right\rceil$;
    allocate grid $\mathscr{G}$ of size $(s_x \times s_y)$;
    for $k = 0$ to $n - 1$ do
        $\begin{bmatrix} i_k \\ j_k \end{bmatrix} = \left\lfloor \frac{1}{r} \left( \begin{bmatrix} x_k \\ y_k \end{bmatrix} - \begin{bmatrix} x_{min} \\ y_{min} \end{bmatrix} \right) \right\rfloor$;
        $\mathscr{G}[i_k][j_k]$.insert($k$);
    return $\mathscr{G}$;
```

The registration algorithms *ICP* and *ARS-Grid* rely on iterative search of closest point. A grid data structure allows such efficient search in the floorplan map with $\mathscr{O}(1)$ complexity. Algorithm 1 illustrates the procedure. The number of cells on each axis must be defined (lines 2-3) to correctly allocate the grid (line 4). Thereafter, the corresponding grid indices $[i_k \quad j_k]^T$ are evaluated for each map point $[x_k \quad y_k]^T$ (line 6) and then each point index k is inserted in its respective grid cell (line 7).

During *ICP* registration, the closest point in the map is found for each laser scan point. Locating the cell of the grid containing the laser scan point is trivial, since it is the same as line 6. A fixed radius centred on the laser scan point defines the grid cells being part of its neighborhood. In this region, the distance between each map point of each cell and the laser scan point is measured and the closest point is detected. The construction of the grid structure is usually very fast so the integration with the virtual scan generation is almost flawless. The grid is allocated once and it is populated when a new virtual scan is available.

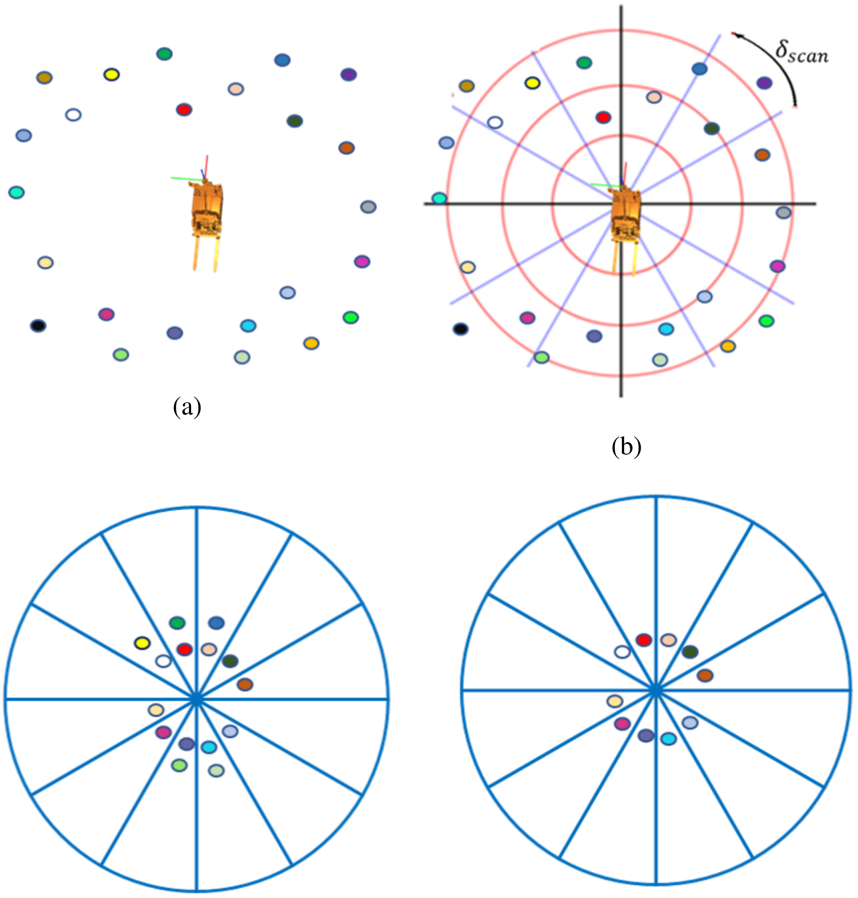
4.3.2 Virtual Scan

A naive approach to localization would perform registration between the current scan and the whole 2D map of the industrial plant. Instead of considering the entire 2D map, a virtual scan is generated for each iteration of *ICP* and *ARS-Grid* registration. A more effective procedure computes the *virtual scan* corresponding to the map region observed from a given viewpoint. The registration process is performed between the laser scan and the virtual scan more efficiently and limiting ambiguity due to environment symmetries. Algorithm 2 and Figure 4.2 present the procedure for the computation of virtual scan.

Using the grid search defined in 4.3.1, the points further than the maximum laser scanner range are discarded to create $\mathscr{M}$ (line 2). Each virtual scan point is initialized to $+\infty$ (line 3). Then the suitable Cartesian map points p are converted into polar coordinates (line 5) and next a parametrization of the angle is applied (line 6). Finally, for each angular sector only the closest point from the scan centre c_0 is inserted into the virtual scan (lines 7-8). The virtual scan is formed by range measurements so a trivial conversion to Cartesian coordinates is required when performing registration,

Algorithm 2 Virtual Scan Computation

```
Inputs: $\mathscr{P} = \{p_k\}_{k=0,\dots,n-1}$ with $p_k = [x_k\ y_k]^T$: map points,
        $d_{max}$: maximum laser scanner range,
        $r$: laser scanner angular resolution,
        $c_0 = [x_c, y_c]^T$: virtual scan centre,
        $\mathscr{G}$: 2D grid structure,
Output: $\mathscr{V}_{\mathscr{S}} = s[v]_{v=0,\dots,m-1}$: generated virtual scan ranges,
procedure VIRTUAL SCAN($\mathscr{P}$)
    $\mathscr{M} = \{p_k \in \mathscr{P} : ||p_k - c_0|| < d_{max}\}$ using $\mathscr{G}$ ;
    $\forall v \in \mathscr{V}_{\mathscr{S}} : s[v] = +\infty$ ;
    $\forall p \in \mathscr{M}$ :
        $\rho_k = \sqrt{x_k^2 + y_k^2}, \quad \theta_k = atan2(y_k, x_k)$ ;
        $i_k = \lfloor \theta_k / r \rfloor$ ;
        if $\rho_k < s[i_k]$ then
            $s[i_k] = \rho_k$ ;
    return $\mathscr{V}_{\mathscr{S}}$;
```



(a)

(b)

(c)

(d)

Figure 4.2: Visualization of Virtual Scan Computation algorithm 2; (a): candidate points; (b): definition of angular sectors and maximum radius; (c): identification of the group of points in each sector; (d): identification of the closest point in each sector.

since the real laser scan points are returned in $[x \quad y]^T$ format. The execution of this algorithm takes about few milliseconds and can be performed for each iteration of the localization procedure.

4.4 Sensor Processing

4.4.1 Localization with SKIP

Feature detection from raw laser scans avoids the dependence of AGV localization from special landmarks. Unfortunately, the extraction of stable and distinguishable interest points from laser measurements in cluttered environments is not a trivial task. In a previous work [57] we proposed FALKO features for AGV localization with laser scanner. FALKO features are extracted where there is a distinctive local distribution of neighbour points. While this criterion succeeds in finding reasonably stable references for navigation and local mapping, the obtained interest points do not describe the global shape of the free space in a laser scan.

Thus, we developed the novel feature SKIP, which extracts interest points through simplification of the general contour of scan points. Algorithm 3 illustrates SKIP detection, which is inspired by the curve simplification technique in [72]. The input data consists of a single laser scan with field-of-view of 360 *deg* represented as a curve connecting adjacent points $\mathscr{P} = \{p_0, \dots, p_{n-1}\}$. Each point p_i has a previous point prv_i and a next nxt_i in a circularly linked list representing the scan. The curve is split in correspondence to gaps caused by occlusion. A gap is marked when the difference between consecutive range measurements is above a threshold (lines 3-6). A cornerness score $score_i$ is computed for each of the valid points p_i based on its previous and next points, respectively p_{i-1} and p_{i+1}. Then, the points are pushed into a priority queue with the minimum $score_i$ on the top position (line 9). Thus, the points with a score less than threshold thr_s are removed from the queue (lines 11-19). When a point is removed from the queue, the score of its previous and next points, respectively prv_i and nxt_i, must be recomputed, since their neighbours are changed. Hence, the prv_i and nxt_i are marked as non-valid (lines 14-15) and their score will be recomputed when extracted from the queue (lines 17-19). The points in the queue with score above threshold thr_s are labelled as feature keypoints (lines 21-23).

SKIP keypoints can be matched with the landmarks belonging to a previously defined map and used in EKF localization based on the standard landmark sensor model instead of the model illustrated in Section 4.2. An example of the outcome of SKIP

Algorithm 3 SKIP Detection

```
Inputs: $\mathscr{P} = \{p_i\}_{i=0,\dots,n-1}$: points of a laser scan,
        $\{r_i\}_{i=0,\dots,n-1}$: ranges of the scan ($r_i = \|p_i\|$),
        $(m_g, q_g)$: gap threshold parameters,
        $thr_s$: splitting threshold,
Output: $\mathscr{F} = \{f_j\}$: SKIP keypoints
procedure SkipDetection($\mathscr{P}, m_g, q_g, thr$)
    for $p_i \in \mathscr{P}$ do
        $thr_{prv} \leftarrow q_g + m_g \frac{r_i + r_{i-1}}{2}$, $thr_{nxt} \leftarrow q_g + m_g \frac{r_i + r_{i+1}}{2}$;
        if $|r_i - r_{i-1}| < thr_{prv}$ or $|r_{i+1} - r_i| < thr_{nxt}$ then
            State $score_i \leftarrow \infty$,
            $val_i \leftarrow false$;;
        else
            $score_i \leftarrow \|p_{i+1} - p_i\| + \|p_i - p_{i-1}\| - \|p_{i+1} - p_{i-1}\|$;
            push $(i, p_i)$ in priority queue $\mathscr{Q}$;
            $prv_i \leftarrow i - 1$, $nxt_i \leftarrow i + 1$, $val_i \leftarrow true$;
    while $\mathscr{Q} \neq \emptyset$ and $score_{top(\mathscr{Q})} < thr_s$ do
        pop $(i, p_i)$ from $\mathscr{Q}$;
        if $val_i$ is true then
            $nxt_{prv_i} \leftarrow nxt_i$, $val_{prv_i} \leftarrow false$;
            $prv_{nxt_i} \leftarrow prv_i$, $val_{nxt_i} \leftarrow false$;
        else
            $score_i \leftarrow \|p_{nxt_i} - p_i\| + \|p_i - p_{prv_i}\| - \|p_{nxt_i} - p_{prv_i}\|$;
            $val_i \leftarrow true$;
            push $(i, p_i)$ in priority queue $\mathscr{Q}$;
    $\mathscr{F} \leftarrow \emptyset$;
    while $\mathscr{Q} \neq \emptyset$ and $score_{top(\mathscr{Q})} < thr_s$ do
        pop $(j, f_j)$ from $\mathscr{Q}$;
        $\mathscr{F} \leftarrow \mathscr{F} \cup \{f_j\}$;
    return $\mathscr{F}$;
```

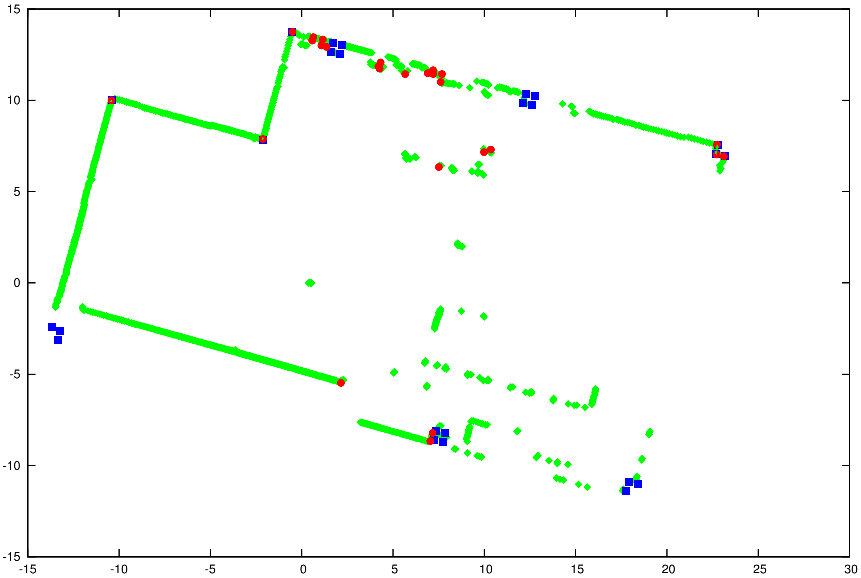


Figure 4.3: An examples of SKIP keypoint features (red circles) associated to landmarks (blue squares) for robot localization. In this case 8 keypoints are correctly associated.

algorithm as well as a landmark map is illustrated in Figure 4.3 The map can be easily extracted from a TLS survey of the industrial plant, collecting the desired corners and their coordinates at the specific laser sensor height. Features detected from laser scans could potentially substitute the reflective artificial landmarks currently used in commercial localization systems.

4.4.2 Localization with ICP Registration

Iterative Closest Point (*ICP*) is one of the mainstream algorithm for point-cloud registration, both 2D and 3D. Given two point sets $Q = \{q_1, \ldots, q_n\}$ and $P = \{p_1, \ldots, p_n\}$, it aims to find the translation t and the rotation R that minimizes the error function defined as

$$E(R,t) = \frac{1}{n}\sum_{i=1}^{n} ||q_i - R\, p_i - t||^2 \tag{4.5}$$

where q_i and p_i are corresponding points, namely pair of points determined by a point q_i and its closest point in $P, \forall q_i \in Q$. So each point in Q is paired with its closest point in P and each pair gives its contribution in the minimization of the error function, exploiting the (R,t) transformation that better aligns the two point sets.

Many different *ICP* variants have been presented since its introduction in [73]. The core behaviour of the method is well known therefore the complete algorithm is not reported. As stated in 4.1, our *ICP* method matches the actual laser scan and the virtual scan generated with a grid structure, as described in Section 4.3. Performing registration in such a way is fast and reliable; furthermore, the *ICP* estimation can be interrupted after a certain period of time, returning a pose estimation even if the convergence condition is not met. This is a remarkable feature of the approach since many other localization methods cannot be stopped until the termination condition is encountered. Conversely, the *ICP* estimation can offer a trustworthy or partial information, depending on the time elapsed and the geometric conditions.

4.4.3 Localization with Correlative Registration

The Correlative Registration approach allows to find the pose $\xi^L = [t_\xi, \theta_\xi]$ of (4.4) by combining together the following two methods:

- We first use the Angular Radon Spectrum method (ARS) presented in [71] to obtain an estimation of the rotation angle θ_ξ.

- Once the current scan points are rotated by the orientation angles θ_ξ, using a grid representation of submaps and current scan points, we find an estimation of t_ξ via $2D$ Grid-based Matching [74, 75].

ARS Correlation

Following [71] we use the Angular Radon Spectrum to characterize the orientation of the scan points by measuring its alignment with respect to a pencil of parallel lines. In particular in [71] it is shown how, given a Gaussian Mixture Model (GMM) representing the points distribution, one can define the ARS as a continuous function derived from the Radon Transform of such distribution. Moreover, approximating the value of the GMM-ARS by a Fourier series expansion, the following equation is obtained:

$$\mathscr{S}_{\kappa}\left[f\right]\left(\theta\right) = a_0 + \sum_{k=1}^{+\infty} \left(a_k \cos(2k\theta) + b_k \sin(2k\theta)\right) \tag{4.6}$$

where a_k and b_k are the coefficients of the series (see [71] for details). Let $\mathscr{S}_{\kappa}\left[f_S\right]\left(\theta\right)$ and $\mathscr{S}_{\kappa}\left[f_D\right]\left(\theta\right)$ be the GMM-ARSs of respectively the source and the destination point sets, then the rotation angle between them can be estimated through the correlation of the corresponding spectra. Let $\{a_k^S, b_k^S\}_k$ and $\{a_k^D, b_k^D\}_k$ be the respective Fourier series coefficients, their correlation is given by

$$C(\delta) = a_0^C + \sum_{k=1}^{+\infty} \left(a_k^C \cos(2k\delta) + b_k^C \sin(2k\delta)\right) \tag{4.7}$$

where $a_k^C = \frac{a_k^D a_k^S + b_k^D b_k^S}{2}$ and $b_k^C = \frac{a_k^D b_k^S - b_k^D a_k^S}{2}$. Then, the rotation angle between source and destination point sets corresponds to the global maximum of the ARS correlation (4.7) [71].

Grid Correlation

Different methods exist to approximate a planar points set using occupancy grids.

In this Section we use the occupancy grid:

$$M^H : r\mathbb{I} \times r\mathbb{J} \rightarrow \{0, 1\} \tag{4.8}$$

with resolution $r \in \mathbb{R}$, $\mathbb{I} = \{i\}_{i=0,\dots,I}$ and $\mathbb{J} = \{j\}_{j=0,\dots,J}$, which map from discrete grid points to 0 and 1 values. Let $m_{i,j}^H : (i, j) \in \mathbb{I} \times \mathbb{J}$ be an element of M^H, we assign it

to zero if no points from set H fall in interval $[ir+\frac{rI}{2}, ir+\frac{rI}{2}+r]\times[jr+\frac{rJ}{2}, jr+\frac{rJ}{2}+r]$ and to 1 if at least one point lies in such interval.
Then we are interested in finding the optimal match:

$$t_\xi^* = \underset{t_\xi \in W}{\text{argmax}} \sum_{i=0}^{I}\sum_{j=0}^{J} m_{ij}^{S} m_{ij}^{t_\xi, D}, \quad I, J \in \mathbb{N}, \tag{4.9}$$

where m_{ij}^{S} are the grid values obtained from a source points set $S \subset \mathbb{R}^2$ and $m_{ij}^{t_\xi, D}$ are the grid values obtained from a destination points set $D \subset \mathbb{R}^2$, after the translation t_ξ is applied on each point of D. Moreover, W is the search window. Following [75] we compute an integral number of steps to cover W as follows:

$$w_x = \left\lceil \frac{W_x}{r} \right\rceil, \quad w_y = \left\lceil \frac{W_y}{r} \right\rceil.$$

In this way, we obtain the following finite set for W:

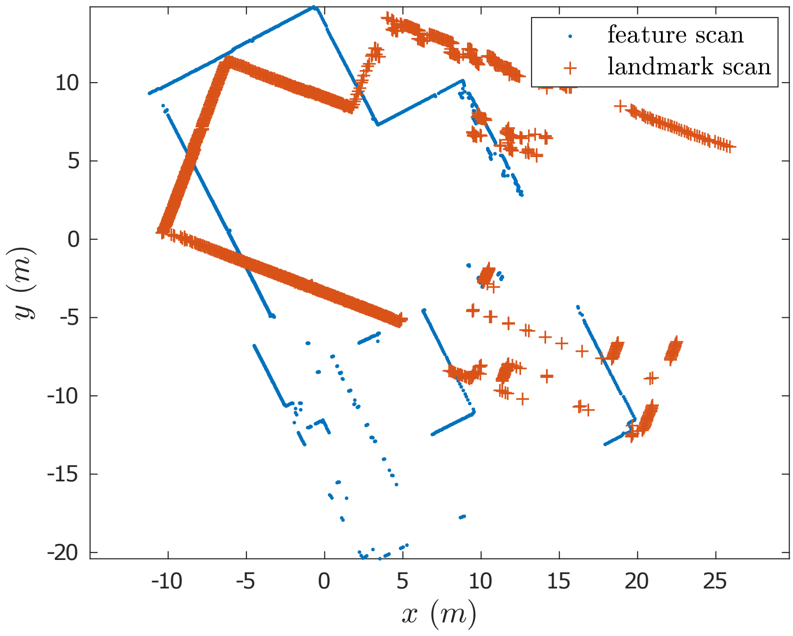
$$\overline{W} = \{-w_x, \ldots, w_x\} \times \{-w_y, \ldots, w_y\},$$
$$W = \{t_{\xi_0} + (rj_x, rj_y) : (j_x, j_y) \in \overline{W}\}.$$

which corresponds to a search window centred in the estimated translation $t_{\xi_0} = [\xi_{x0}, \xi_{y0}]$. In our application we get this initial guess by the EKF current state.

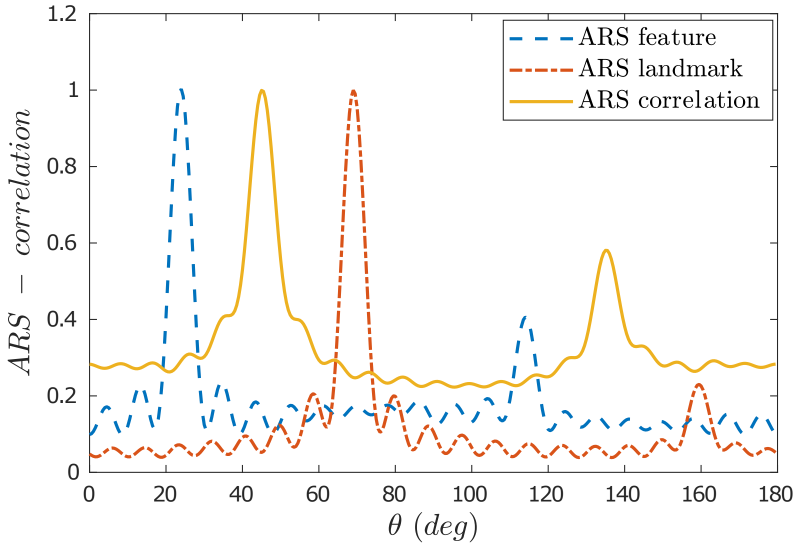
In order to find the best match that solves problem (4.9), we use Algorithm 4 which consists in element-wise multiplications between matrices. Actually, more efficient algorithms exist to do it. For instance, [75] uses a branch and bound approach to efficiently compute ξ^* even for larger search windows (there, also the orientation angle is considered). In order to find a faster solution, following [76] we implement a multi-grid map with two levels of resolution. Here the strategy is to use a low-resolution map to quickly identify areas that might contain the global maximum. By doing so, it is possible to minimize the volume that is searched at high resolution. In particular, we use a factor of 10 between low and high resolution. This choice allows us to use search windows $W_x = W_y = 0.5$ *m* and high resolution $r_h = 1$ *cm* (low resolution $r_l = 10$ *cm*) with a computational time below 100 *ms*.

Algorithm 4 Grid Correlation

```
Inputs: $m_{ij}^S$: map of source points set,
        $m_{ij}^D$: map of destination points set,
        $W = [W_x, W_y]$: search window,
        $t_{\xi_0} = [\xi_{x_0}, \xi_{y_0}]$: initial guess on translation.
Output: $t_\xi^*$: estimated translation.
procedure GridCorrelation($m_{ij}^S, m_{ij}^D, W, t_{\xi_0}$)
    $best\_score = -\infty$
    for $s_x = -w_x$ to $w_x$ do
        for $s_y = -w_y$ to $w_y$ do
            $\overline{t_\xi} = t_{\xi_0} + (rs_x, rs_y)$
            $score = \sum_{i=0}^{I} \sum_{j=0}^{J} m_{ij}^S \; m_{ij}^{\overline{t_\xi},D}$
            if $score > best\_score$ then
                $t_\xi^* = \overline{t_\xi}$
                $best\_score = score$
    return $t_\xi^*$
```

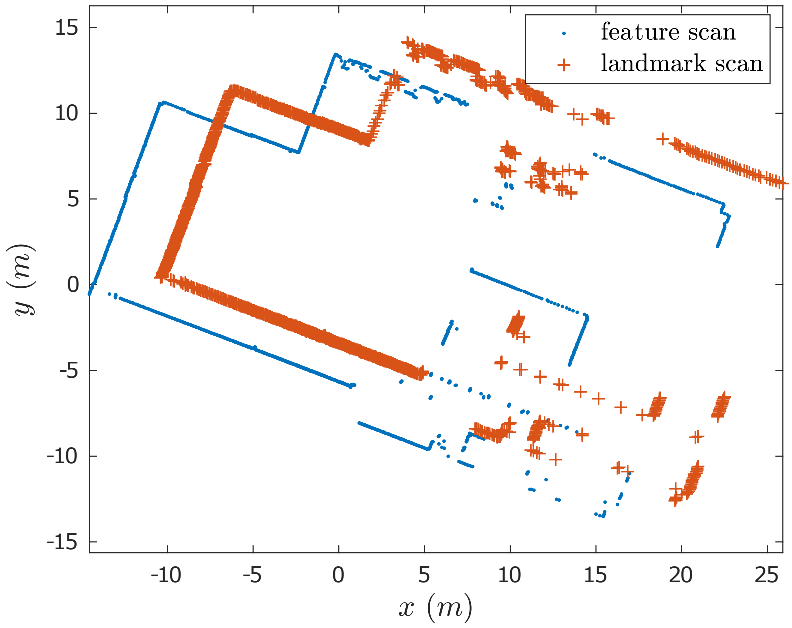


(a)

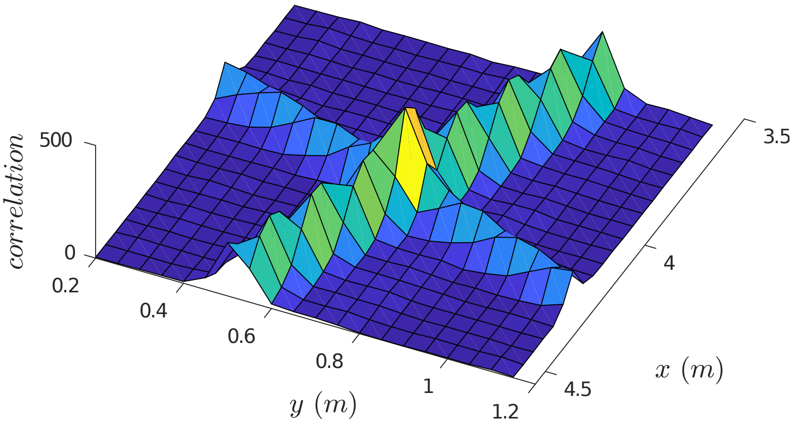


(b)

Figure 4.4: Principal steps of the correlative registration process. (a) submap and current scan; (b) resulting ARS correlation.

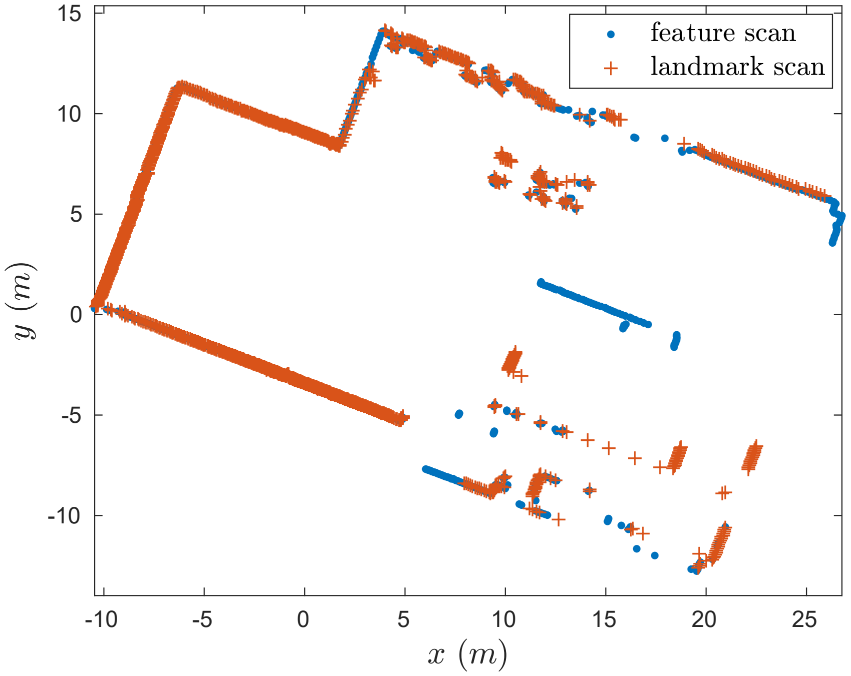


(c)



(d)

Figure 4.5: Principal steps of the correlative registration process. (c) current scan rotated by the ARS correlation angle; (d) resulting grid correlation using windows size $W_x = W_y = 0.5\ m$.



(e)

Figure 4.6: Principal steps of the correlative registration process. (e) final result of the procedure.

Correlative Registration

The combination of ARS and Grid-based Matching approaches needs to transform the submap $H^{\hat{V}}$ into the data set:

$$\mu = \left\{ m_{ij}^{H^{\hat{V}}}, \{ a_k^{H^{\hat{V}}}, b_k^{H^{\hat{V}}} \} \right\},$$

where $m_{ij}^{H^{\hat{V}}}$ represents the grid values of (4.8) obtained from submap $H^{\hat{V}}$. Moreover, $\{ a_k^{H^{\hat{V}}}, b_k^{H^{\hat{V}}} \}$ are the coefficients of the Fourier series $\mathscr{S}_\kappa \left[f_{H^{\hat{V}}} \right] (\theta)$ which approximates the GMM-ARS for $H^{\hat{V}}$. Given the submap data set μ, a current scan H^V, a search window W and a guess pose ξ_0, Algorithm 5 returns the estimated pose ξ^* of the current scan points in the submap frame. In lines 2-5 the angle θ_ξ^*, that represents the rotation between current and submap frames, is computed. Here, first the coefficients of the correlation between the Fourier series $\mathscr{S}_\kappa \left[f_{H^{\hat{V}}} \right] (\theta)$, $\mathscr{S}_\kappa \left[f_{H^V} \right] (\theta)$ are computed. Then the rotation angle θ_ξ^* is found as the local maximum of the ARS correlation into the interval $S_\theta = [\xi_{\theta_0} - \Delta_\theta, \xi_{\theta_0} + \Delta_\theta]$ centered in ξ_{θ_0}, e.g. $\Delta_\theta = 30°$.

The translation component of ξ^* is computed in lines 6-8. Once the rotation $R_{\theta_\xi^*}$ is applied to the current scan points H^V (line 6), the translation between the rotated current scan points $H^V_{\theta_\xi^*}$ and the submap points $H^{\hat{V}}$ is found by Algorithm 4.

4.5 Setup

As introduced in Section 4.1, the standard AGV control architecture is PLC-based, meeting the *IEC 61131* standard requirements. To increase the performance of the system, a different solution is proposed. The framework relies on an industrial PLC and an industrial x64 PC; the first performs every canonical AGV task including the Kalman filtering and the second only evaluates pose estimation, which is calculated with the techniques described in Section 4.4. Data communication within the framework is socket-based. The AGV uses a laser rangefinder with 360 *degrees* of field-of-view and 3600 points returned in each scan every 120 ms. All the experiments described in the following are performed on an industrial PLC with an Intel Core i7-2610UE @ 1.5 GHz CPU and 2 GB of DDR3 RAM combined with an industrial

Algorithm 5 Correlative Registration

```
Inputs: $\mu = \left\{ m_{ij}^{H^{\hat{V}}}, \{a_k^{H^{\hat{V}}}, b_k^{H^{\hat{V}}}\} \right\}$: submap data set,
        $H^V$: current scan,
        $W = [W_x, W_y]$: search window,
        $\xi_0 = [\xi_{x_0}, \xi_{y_0}, \xi_{\theta_0}]$: guess pose.
Output: $\xi^* = [t_\xi^*, \theta_\xi^*]$: estimated pose.
procedure REGISTRATION($\mu, H^V, W, \xi_0$)
    compute the ARS coefficients $\{a_k^{H^V}, b_k^{H^V}\}$
    compute the ARS correlation coefficients $\{a_k^C, b_k^C\}_{H^{\hat{V}}, H^V}$
    $S_\theta = [\xi_{\theta_0} - \Delta_\theta, \xi_{\theta_0} + \Delta_\theta]$
    $\theta_\xi^* = findMaxCorr\left( \{a_k^C, b_k^C\}_{H^{\hat{V}}, H^V}, S_\theta \right)$
    $H_{\theta_\xi^*}^V = R_{\theta_\xi^*} H^V$
    compute $m_{ij}^{H^V_{\theta_\xi^*}}$
    $t_\xi^* = $ GRIDCORRELATION($m_{ij}^{H^{\hat{V}}}, m_{ij}^{H^V_{\theta_\xi^*}}, W, t_{\xi_0}$)
    return $\xi^*$
```

x64 PC with an Intel Core i7-6820EQ @ 2.6 GHz CPU and 32 GB of DDR4 RAM running C-based code. Experiments are performed with a tricycle AGV calibrated using the method in [18].

4.6 Experiments

We validate the precision of each localization method evaluating the standard deviation of the final AGV pose for halting points A-D of Figure 4.7. In the Figure, the position of the center of the rear wheels axle of the vehicle is represented by a filled circle. The AGV pose is estimated through a Total Station which measures the position of physical points of the AGV in a global reference frame with sub-millimetre accuracy.

Table 4.1 show the standard deviations, along longitudinal and lateral directions, for the proposed methods over 10 repeated trials for each halting point. Figures 4.8, 4.9, and 4.10 shows the final AGV poses of each repeated trial for halting points. We run the *ARS-Grid* method using different values of grid resolution r_h. The grid resolution of 10 *mm* seems to be the best trade off between performance and precision. More in general, all the compared methods have a standard deviation largely below 1 *cm*. *ICP* achieves the lowest standard deviation followed by *Reflectors*, *ARS-Grid* and *SKIP*. The *SKIP* method is sensitive to the number of features detected in the environment and to their position accuracy.

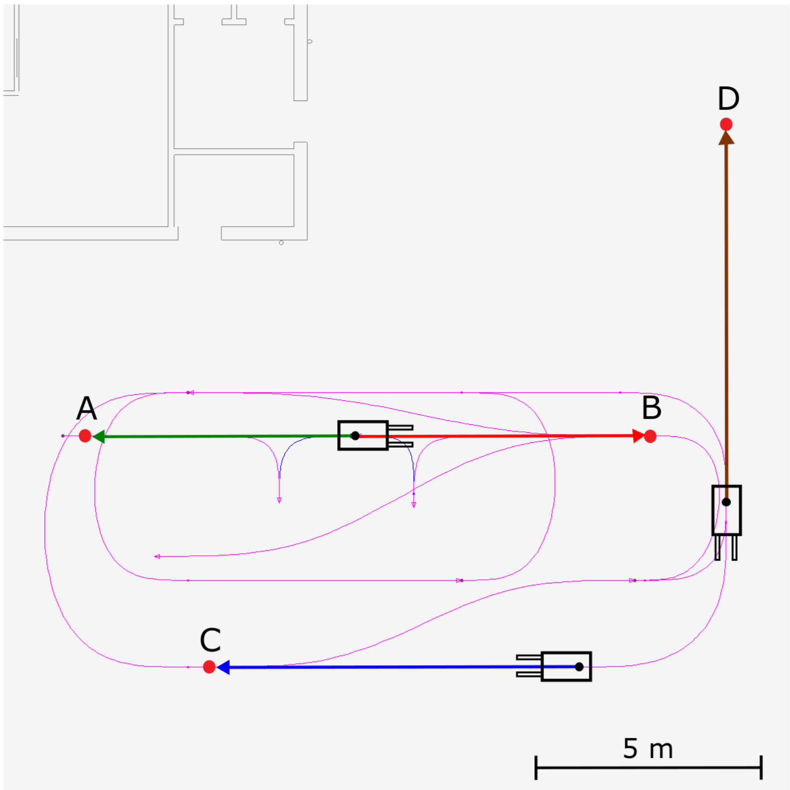


Figure 4.7: Halting points (A, B, C and D) and related paths that we use to test the four localization methods. For each path the starting poses are represented by a miniature of the vehicle (A and B are reached from the same starting pose).

goal	σ_{lon} [mm] , σ_{lat} [mm]													
	ARS-Grid (5 *mm*)		*ARS-Grid* (10 *mm*)		*ARS-Grid* (20 *mm*)		*ARS-Grid* (50 *mm*)		*ICP*		*SKIP*		*Reflectors*	
A	1.60	0.87	1.39	0.79	4.89	1.84	9.23	3.95	0.27	0.16	3.20	0.91	1.47	0.66
B	3.61	1.99	3.64	0.77	3.33	0.66	6.02	2.29	1.07	0.10	4.30	1.04	1.27	1.01
C	4.53	0.51	3.40	0.67	1.99	0.91	10.05	5.35	0.87	0.08	4.73	1.03	1.07	0.48
D	1.39	1.12	1.97	0.61	3.54	2.45	11.42	8.11	0.52	0.34	2.12	1.80	1.65	0.37

Table 4.1: Standard deviations of halting points A-D along lateral and longitudinal directions for the proposed methods.

Furthermore, we analyze the robustness of *ARS-Grid* and *ICP* through two experiments. In the first one we simulate an inaccurate initial guess adding an offset on variables x, y and θ. In the other one we occlude an angle of the laser view to simulate a loss of laser information. In both cases we run the *ARS-Grid* method using a grid resolution of 10 *mm* and search window $W_x = W_y = 0.5$ *m*. The AGV goes through an ellipsoid path of around 35 meters (Figures 4.11, 4.12, 4.13, 4.14). Moreover, we allocate for *ARS-Grid* and *ICP* a maximum of 100 *ms*.

Tables 4.2, 4.3 show the mean error and the mean computational time for these experiments. Here the errors on x, y and θ are evaluated as the difference between the AGV path computed with offset (or laser occlusion) and the path found in ideal conditions (without offset and laser occlusion). The mean time for *ARS-Grid* and *ICP* without offset and laser occlusion is 73.7 *ms* and 59.3 *ms*, respectively. According to the results, *ICP* seems to be sensitive to the initial guess accuracy, especially for the component θ. In fact, when the offset on theta is greater than 5 degrees, the paths found by *ICP* differ a lot from the path found with no offset (Figures 4.11, 4.12, 4.13). On the other hand, *ARS-Grid* seems to be more sensitive to the loss of laser information (Figure 4.14).

Thus, *ICP* seems to be the best approach as long as the initial guess on the current AGV pose is sufficiently close to the correct value. Such condition is always satisfied in EKF localization systems, which frequently updates the estimation. Conversely, *ARS-Grid* enables global registration and could be used when no accurate initial guess is available.

	$offset$: x,y [mm], θ [deg]																	
	[40] [5]		[40] [10]		[40] [15]		[80] [5]		[80] [10]		[80] [15]		[120] [5]		[120] [10]		[120] [15]	
	AG	ICP	AG	ICP	AG	ICP	AG	ICP	AG	ICP	AG	ICP	AG	ICP	AG	ICP	AG	ICP
error x [mm]	7.14	0.87	6.89	164	6.44	178	6.67	0.79	6.61	164	6.67	183	6.61	0.74	6.33	162	6.45	203
error y [mm]	4.96	0.40	4.50	127	4.37	170	4.31	0.35	4.62	124	4.57	167	4.56	0.32	4.55	124	4.45	170
error θ [deg]	0.04	0.01	0.04	4.82	0.04	11.3	0.05	0.01	0.05	4.79	0.05	11.2	0.04	0.01	0.04	4.68	0.04	11.0
time [ms]	73.7	100	73.9	100	73.3	81.6	72.4	100	72.7	100	72.1	83.2	72.1	100	72.0	100	71.9	85.0

Table 4.2: Mean error and mean computational time for *ARS-Grid* (AG) and *ICP* for different offset applied to the initial guess.

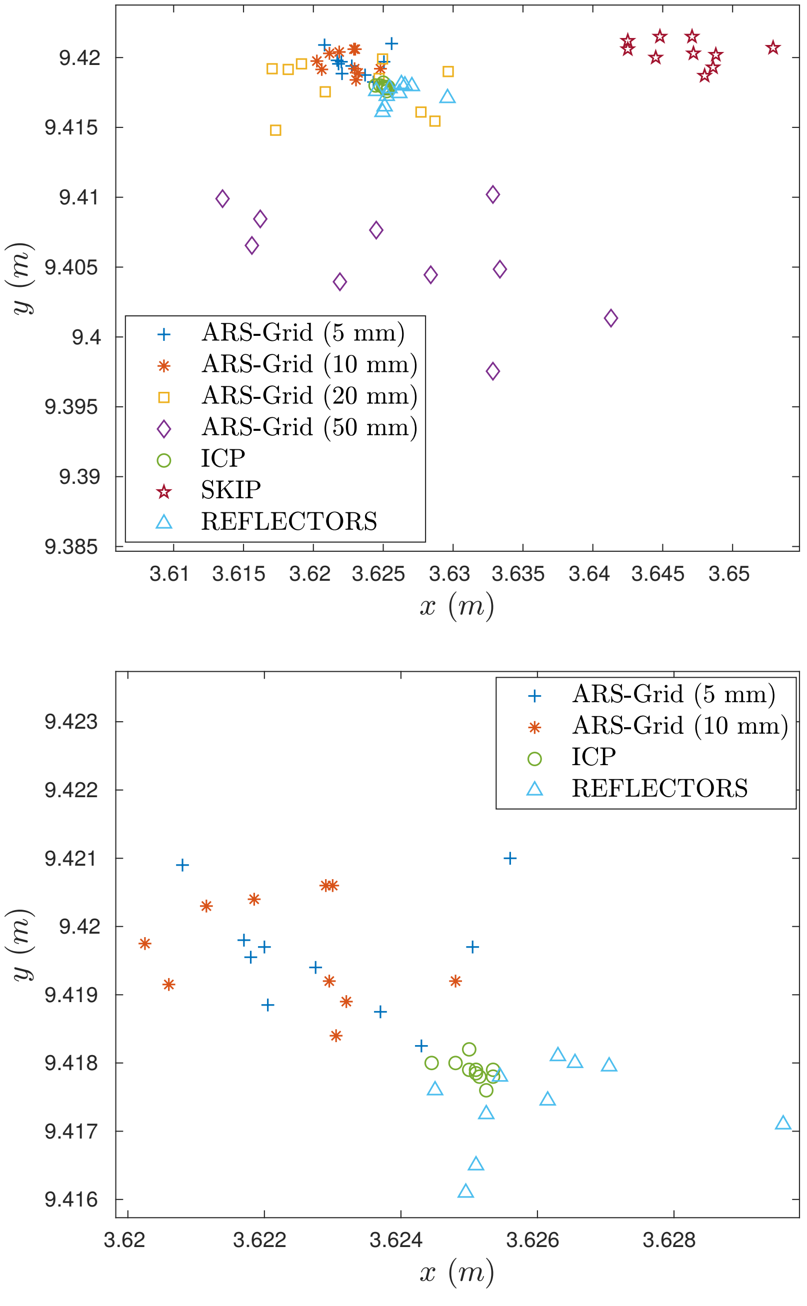
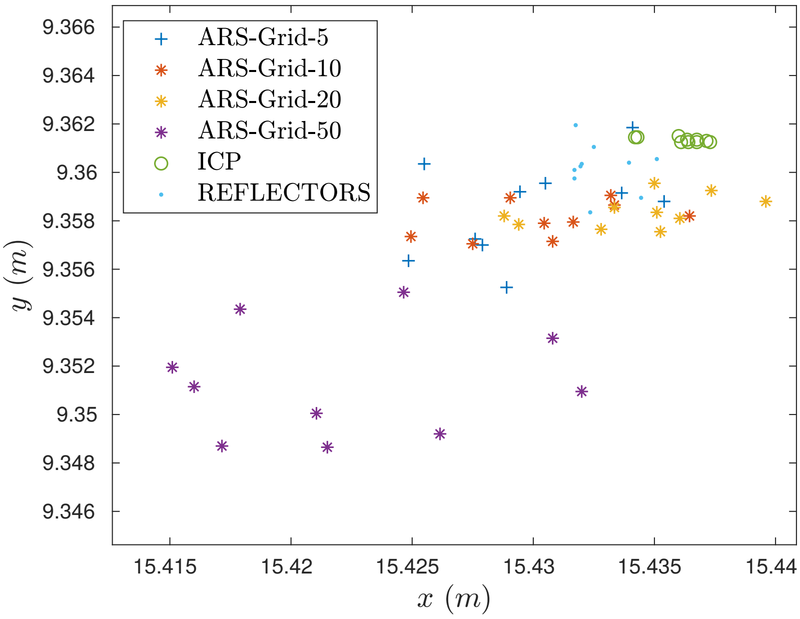


Figure 4.8: Final poses for halting point A for the proposed methods. The image below corresponds to a zoom of the image above without depicting the halting points for *SKIP* and *ARS-Grid* with 20 and 50 *mm* of grid resolution.



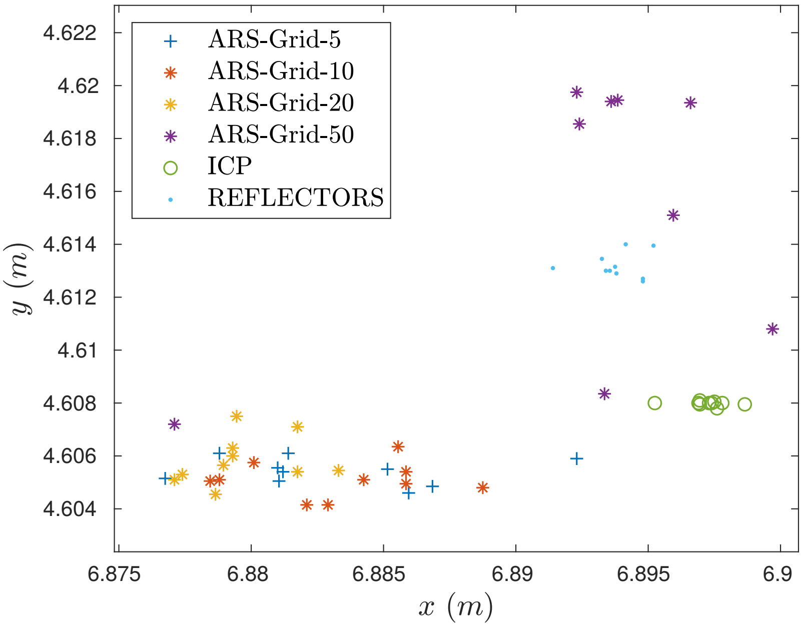
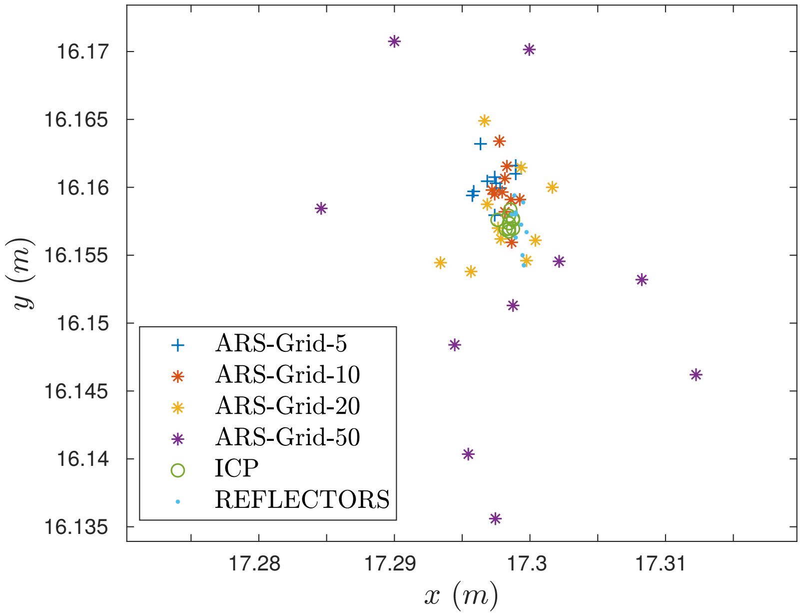


Figure 4.9: Top: Final poses for halting point B for the proposed methods. Bottom: Final poses for halting point C for the proposed methods.



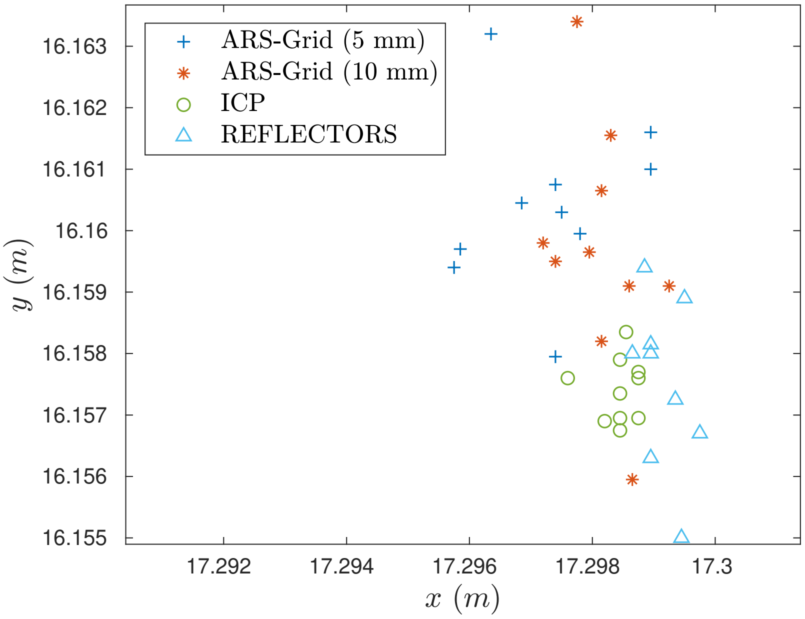
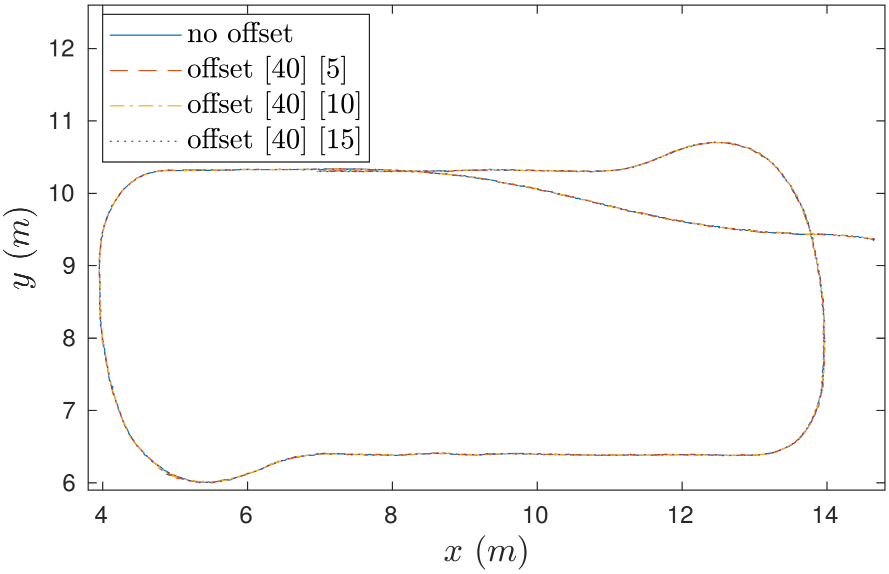


Figure 4.10: Final poses for halting point D for the proposed methods. The image below corresponds to a zoom of the image above without depicting the halting points for *SKIP* and *ARS-Grid* with 20 and 50 *mm* of grid resolution.



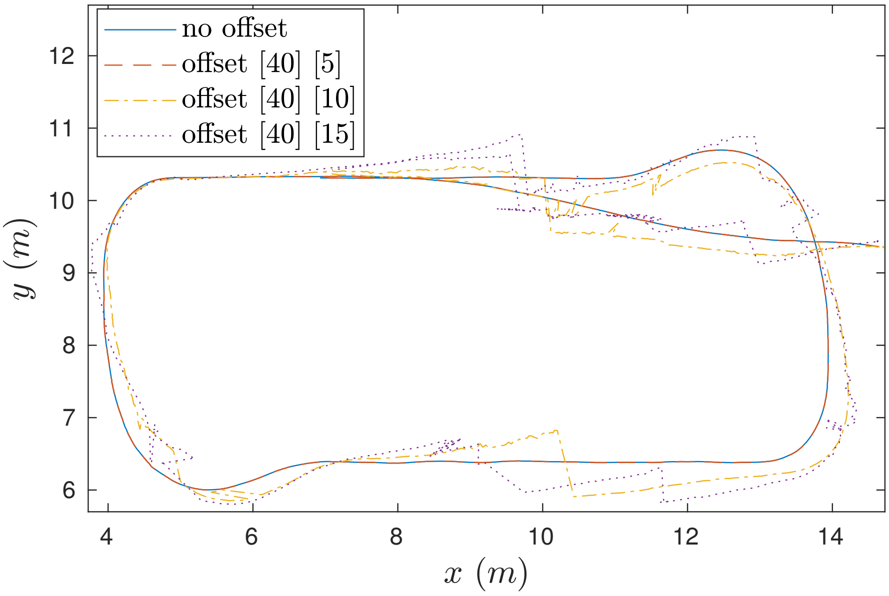
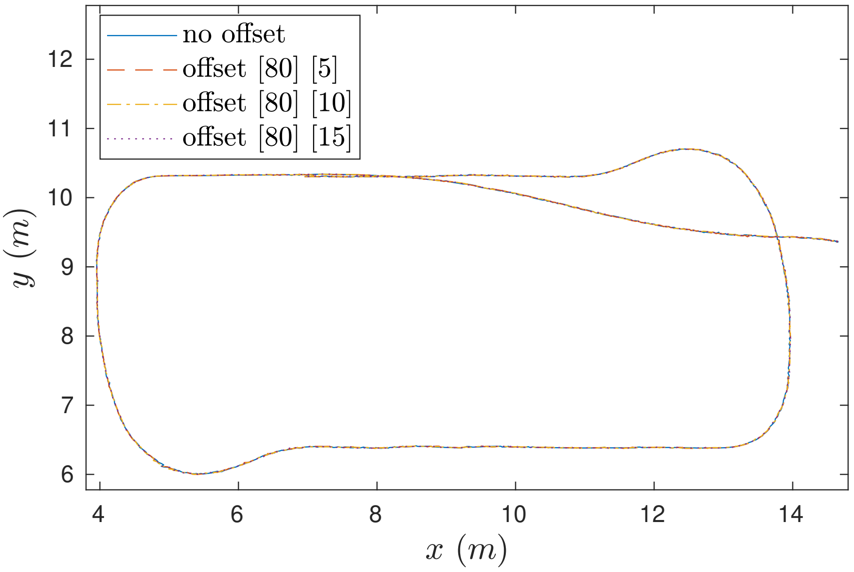


Figure 4.11: Comparison of the AGV paths using offset [40],[5-10-15]. Top: *ARS-Grid*. Bottom: *ICP*.



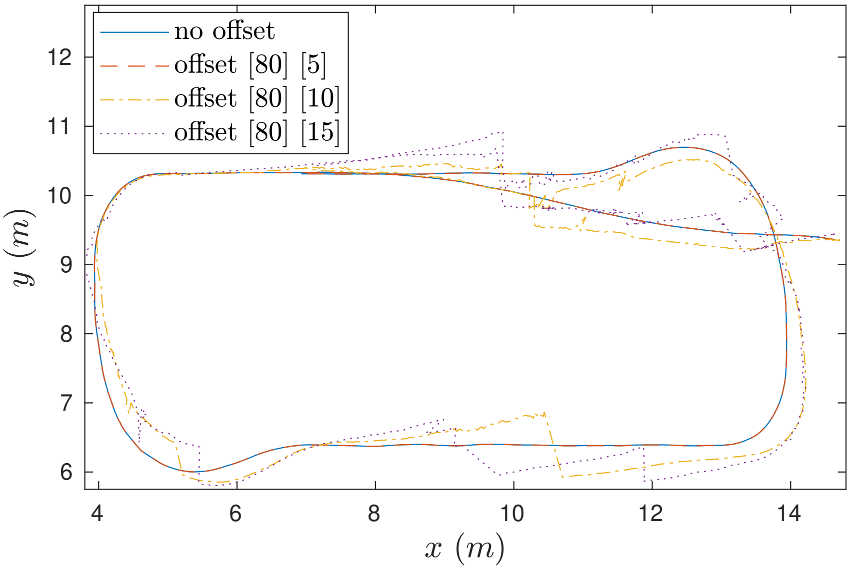
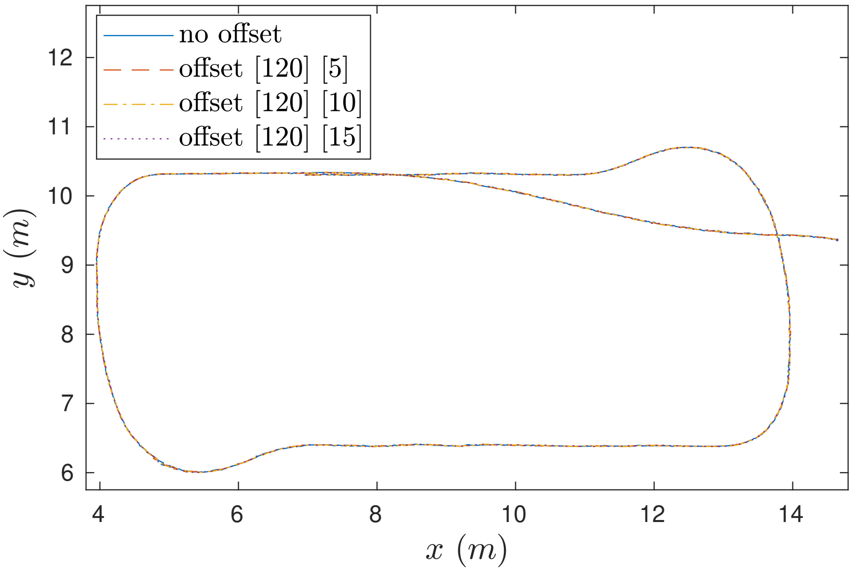


Figure 4.12: Comparison of the AGV paths using offset [80],[5-10-15]. Top: *ARS-Grid*. Bottom: *ICP*.



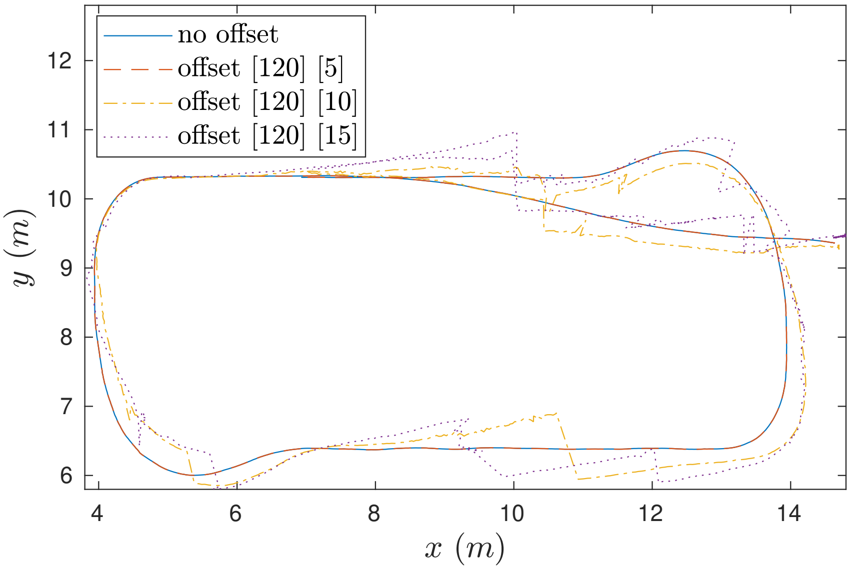
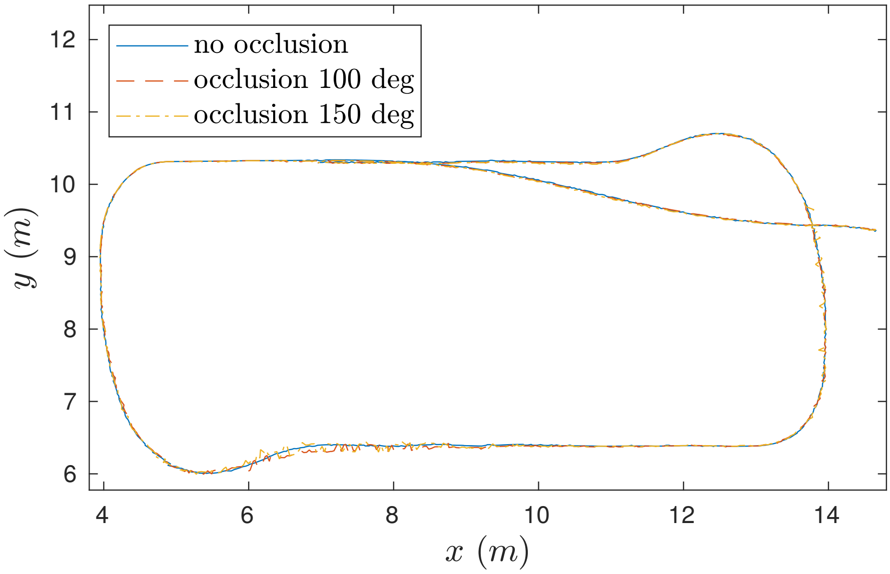


Figure 4.13: Comparison of the AGV paths using offset [120],[5-10-15]. Top: *ARS-Grid*. Bottom: *ICP*.



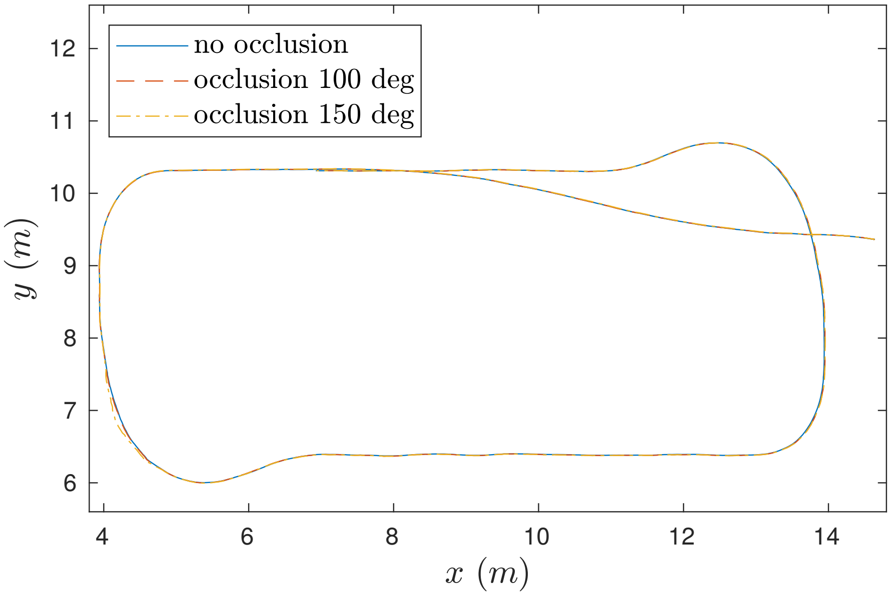


Figure 4.14: Comparison of the AGV paths with occlusion. Top: *ARS-Grid*. Bottom: *ICP*.

	laser occlusion [deg]			
	100		150	
	AG	ICP	AG	ICP
error x [mm]	16.8	3.65	18.3	6.51
error y [mm]	11.9	1.76	11.5	3.72
error θ [deg]	0.21	0.03	0.23	0.06
time [ms]	50.6	40.7	40.6	30.6

Table 4.3: Mean error and mean computational time for *ARS-Grid* and *ICP* with laser occlusions of 27.78 % and 41.67% (100 and 150 degrees).

4.7 Discussion

This chapter has presented three different methods for laser-based AGV localization and assessed their performance in real industrial setup. The proposed approach corresponds to sensor models for EKF localization. The *SKIP* model exploits detection of high curvature keypoint features as landmarks whereas *ICP* and *ARS-Grid* perform registration of the current scan with a dense map of the environment. Exploiting a complete TLS industrial survey, a map of the industrial plant is provided, considering corners (for *SKIP* feature matching) or a thin slice of the 3D point cloud at laser sensor height (for *ICP* and *ARS-Grid* registrations). *ICP* and *ARS-Grid* exploit a virtual scan approach to speed up pose evaluation and avoid localization mismatches.

These methods are compared with standard landmark-based localization using reflectors. The experimental results show that the localization precision of all the methods is less than 1 *cm* and it is compliant with usual industrial requirements. *ICP* generally achieves higher precision whereas *ARS-Grid* with proper grid resolution is mostly comparable with *Reflectors*. Conversely, *SKIP* localization is more sensitive to the number of observed features and to their position accuracy. Additionally, *ICP* pose estimation can return valid results even if interrupted after a maximum time threshold.

Chapter 5

Signature-based Place Recognition

This chapter illustrates Geometric Relation Distribution (GRD), a novel signature for place recognition and loop closure with landmark maps, such as the reflectors used in many commercial localization systems or the SKIP features presented in the previous chapter. GRD encodes geometric pairwise relations between landmark points into a continuous probability density function.

Place recognition is required when no a priori information about robot pose is available or the estimation is not accurate enough. The first case occurs during the initialization of the EKF-based localization, e.g. when the AGV is switched on or when it is inserted in the system for the first time. The latter case corresponds to the loop closure required to build consistent maps during the SLAM procedure. The mapping algorithm checks whether the currently observed region of the environment has been already visited during an arbitrary trajectory and, if this is the case, the accumulated error is spread over the whole path to guarantee accurate localization in every position. The effectiveness of the proposed method is assessed through experiments with standard datasets.

5.1 Introduction

An important task in robot localization and mapping is the recognition of places in the available map. During *global localization* the pose of the robot is initially unknown and the current sensor measurements are compared with all potential matching locations. Similarly, during *SLAM* tasks the robot checks if the current location has been already visited. When the recognition takes place after the robot has travelled long paths, this operation is called *loop closure*. Loop closure enables to recover localization errors and map inconsistencies due to cumulative odometry error by estimating the relative transformation between the current robot frame and the matching one. While this problem has been extensively investigated for several sensor and map models like occupancy grid maps from range finders or computer vision keypoint features, robust loop closure techniques for landmark maps have been developed quite recently [57, 77–80].

The keypoint features [79, 81] detected in range finder scans naturally constitute local maps of landmarks. The landmark coordinates of each local map are referred to the corresponding local keyframe. Given this formulation, global localization or loop closure are achieved by associating a query landmark set and all the stored landmark sets in the global map. Direct landmark-to-landmark association is neither computationally efficient nor reliable over large collections of points. Thus, most loop closure methods perform point-to-point matching, after selection of candidates by comparing *signatures*. Several signatures [77, 80] are designed on bag-of-word (BoW) and explicitly require features with descriptors. BoW approaches rely on descriptors and their performance is affected when descriptors poorly characterize landmark neighbourhoods. Signatures like GLARE [78] and GLAROT [57] encode the geometric relations between the features like pairwise distances and relative angles into histograms. In particular, GLAROT is invariant to translation and rotation in order to detect visited submaps regardless of viewpoint. These signatures capture distinctive distribution of feature points and do not require neither specific descriptors nor preliminary dictionary as for BoW approaches. However, their histogram-based representation is sensitive to the resolution of the discretized geometric parameters.

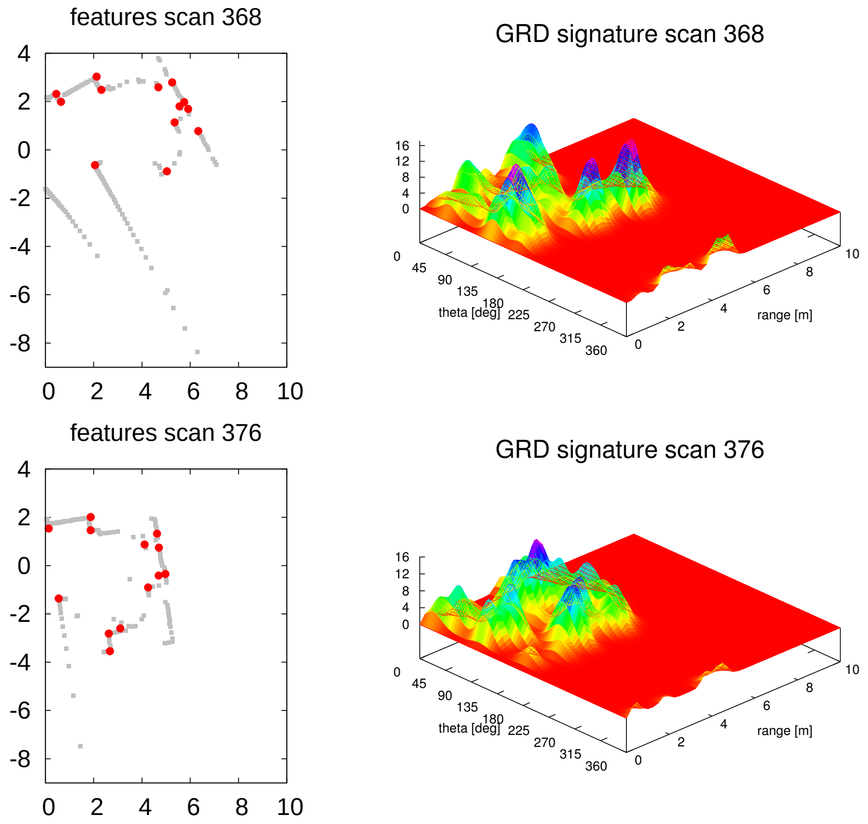


Figure 5.1: Examples of GRD signature. Left: the scans 368 and 376 of dataset *intel-lab* (grey) and the SKIP keypoints (red) detected in the respective scan. Right: the corresponding GRD signatures based on biased Rayleigh distribution obtained from each of the two point sets.

In this chapter, a novel signature *Geometric Relation Distribution* (GRD) for place recognition and loop closure detection in landmark maps is proposed. Like GLARE and GLAROT, GRD encodes geometric relations between landmarks, but its representation corresponds to the probability density function (PDF) of pairwise angles and distances between features. The PDF is multi-modal with a mode for each pair of features. The PDF mode is the product of two marginal distributions, respectively von Mises for angles and Erlang or biased Rayleigh for distances. Moreover, the GRD distribution is expanded into a series of orthogonal functions based on Fourier series and Laguerre polynomials. The truncated series can effectively incorporate arbitrary number of modes using the same basis in order to scale with arbitrary number of features. The series is used to efficiently compute a rotation-invariant correlation metric. The performance of GRD has been assessed through experiments with standard benchmark datasets and compared with the state-of-the-art signatures and loop closure detection algorithms.

The chapter is organized as follows. Section 5.2 reviews the state of the art of loop closure algorithms for landmark maps. Section 5.3 illustrates GRD and its efficient representation with orthogonal function basis. Section 5.4 presents a transformation invariant metric. Section 5.5 reports the experimental results. Section 5.6 concludes the chapter.

5.2 Related Works

The landmark map model is based on the extraction of point features from sensor measurements. Paradoxically, the development of robust keypoint features for planar range finders is quite recent acquisition. Tipaldi et al. [81] proposed FLIRT, the first keypoint feature using multiscale approach and descriptors inspired by computer vision features. FALKO features [79] enable detection of more stable keypoints based on careful neighbourhood selection and cornerness score. FALKO keypoints also support descriptors for encoding neighbour point distribution in order to improve data association and, possibly, loop closure.

Several loop closures algorithms for matching feature-based maps have been developed. The place-recognition problem must be solved in order to identify already visited poses. Loop candidates must be selected using a robust signature to avoid recognition error. Gutmann et al. [82] proposed Local Registration and Global Correlation (LRGC), an incremental method for reconstruction of global maps from range data. Geometrical FLIRT phrases (GFPs) [77] were proposed as an efficient and reliable place recognition exploiting FLIRT descriptors and the bag-of-words (BoW) approach. Deray et al. [80] improved BoW association using a Viterbi algorithm and weak geometric check achieving better loop closure performance. Geometric LAndmark RElations (GLARE) [78, 83] have been proposed to transform 2D laser scans into pose invariant histogram representations evaluating invariant pairwise distances and angles between scan points. Nevertheless, this approach is not orientation invariant. GLAROT [57] was introduced as an improvement over GLARE and designed to be rotation invariant. Point-to-point data association algorithms like maximum clique methods [84, 85] or joint compatibility test [86] are required to check the selected candidates and to estimate the transformation.

A different approach includes the loop closure methods comparing complete scans or local maps with registration algorithms. Lu and Milios [87] proposed the first registration-based loop closure algorithm based on a variant of iterative closest point (ICP). More recent registration methods for loop closure include ICP in pose graphs [88] and efficient alignment of occupancy grid maps [75].

Similar problems are faced when building a map with 3D point clouds. The work in [89] extends GLAROT to space domain through an approximately regular binning of sphere. Tazaki et al. [90] proposed a loop detection method as a pair of partial time series showing high similarity score. Kim et al. [91] introduced a new descriptor for 3D laser scans embedded with an efficient search algorithm based on KD tree online with Ring Key for fast search. Multiple point cloud registration [92] can improve robustness of point-to-point association.

5.3 Pairwise Point-Signature Distribution

Given a set of 2D points $\mathscr{P}$, the pairwise angles and distances of two different points $\mathbf{p}_i, \mathbf{p}_j \in \mathscr{P}$ (assume that $p_{i,y} > p_{j,y}$) are defined respectively as

$$\hat{\theta}_{ij} = \operatorname{atan2}(p_{i,y} - p_{j,y}, p_{i,x} - p_{j,x}) \quad \hat{r}_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\| \tag{5.1}$$

When the input points are acquired through sensors, their coordinates are noisy and uncertain. The point set $\mathscr{P}$ may represent keypoint features extracted from a given robot location. While their Cartesian coordinates are dependent on the viewpoint, their pairwise distances and angles are respectively invariant or invariant up to a rotational offset. Our goal is to define a *signature* using the distribution of pairwise angle θ and distance r on $\mathscr{P}$ disregarding the exact pairs. Each pair (i,j) contributes to each mode $\mathscr{P}_{ij}$ of the global distribution of θ and r. The probability density function (PDF) of $[\theta,r]$ associated to $\mathscr{P}$ is

$$p_{\theta r}(\theta,r) = \sum_{i=1}^{n_p} \sum_{j=i+1}^{n_p} p_{\theta r}(\theta,r|\mathscr{P}_{ij})\ p(\mathscr{P}_{ij}) \tag{5.2}$$

Hence after, we assume that all the modes are equally likely and $p(\mathscr{P}_{ij}) = 1/N$, where $N = n_p(n_p-1)/2$. Moreover, we assume that $p_{\theta r}(\theta,r|\mathscr{P}_{ij})$ can be factorized into the product of the independent PDFs of θ and r as

$$p_{\theta r}(\theta,r|\mathscr{P}_{ij}) = p_{\theta_{ij}}(\theta)\ p_{r_{ij}}(r) \tag{5.3}$$

The two marginal distributions $p_{\theta_{ij}}(\theta)$ and $p_{r_{ij}}(r)$ must be chosen consistently with the random variable domain. The *von Mises distribution* is often used for angular statistics with central mode in preference to the more mathematically cumbersome wrapped normal distribution. Thus, it is used as model for θ with mean values $\hat{\theta}_{ij}$. The probability density function $p_{r_{ij}}(r)$ must represent a random variable on interval $[0,+\infty)$ and single mode in $\hat{r}_{ij}$. There are several models for this purpose and in this chapter we discuss *Erlang* and *biased Rayleigh* distributions. The PDF $p_{\theta r}(\theta,r)$ in Equation (5.3) has the form of weighted sum of uni-modal kernels, which is not convenient for efficient algebraic manipulation. Moreover, a metric invariant to rigid transformation is required to compare two different signatures. The rest of this Section is devoted to address these problems.

5.3.1 Pairwise Angle Statistics with Von Mises distribution

The probability density function of the von Mises distribution is

$$p_{\theta_{ij}}(\theta) = \frac{1}{2\pi \mathrm{I}_0(\kappa)} \exp\left(\kappa \cos\left(\theta - \hat{\theta}_{ij}\right)\right) \tag{5.4}$$

where $\mathrm{I}_0(\kappa)$ is the modified Bessel function of order 0, $\hat{\theta}_{ij}$ is the mean value and also the maximum of density of probability, and κ is the so called concentration parameter, a reciprocal measure of dispersion. We assume that the concentration parameter κ is the same for all the angles. The distribution can be expanded into *Fourier series* [93, (9.6.34)] as

$$p_{\theta_{ij}}(\theta) = \frac{1}{2\pi} + \sum_{k_\theta=1}^{+\infty} \left(a_{k_\theta}^{ij} \cos(k_\theta \theta) + b_{k_\theta}^{ij} \sin(k_\theta \theta) \right) \tag{5.5}$$

$$a_{k_\theta}^{ij} = \frac{\mathrm{I}_{k_\theta}(\kappa) \cos\left(k_\theta \hat{\theta}_{ij}\right)}{\pi \mathrm{I}_0(\kappa)} \quad b_{k_\theta}^{ij} = \frac{\mathrm{I}_{k_\theta}(\kappa) \sin\left(k_\theta \hat{\theta}_{ij}\right)}{\pi \mathrm{I}_0(\kappa)} \tag{5.6}$$

The infinite Fourier series is usually truncated to term n_θ as $s_{n_\theta}(\theta)$. A bound on the approximation error has been derived in [71] as

$$\left| p_{\theta_{ij}}(\theta) - s_{n_\theta}(\theta) \right| \leqslant \frac{\mathrm{I}_{n_\theta}(\kappa)}{\pi \mathrm{I}_0(\kappa)} \frac{\kappa}{n_\theta} \tag{5.7}$$

Such bound largely overestimates the error, but can be used to compute the number of terms n_θ needed to guarantee the required maximum error.

5.3.2 Erlang distribution for Pairwise Distance

The random variable r representing the pairwise distance between two points is defined on domain $[0, +\infty)$ and has single mode corresponding to the measured one. The Erlang distribution is among the probability distributions which satisfy such conditions. Given the scale parameter $\eta_{ij} > 0$ and shape parameter $d_{ij} \in \mathbb{N}$, the PDF of an Erlang distribution is

$$p_{r_{ij}}(r) = \frac{r^{d_{ij}-1} \, \mathrm{e}^{-\frac{r}{\eta_{ij}}}}{\eta_{ij}^{d_{ij}} \, (d_{ij}-1)!} \tag{5.8}$$

The Gamma distribution is a generalization of Erlang where parameter d_{ij} can be any positive real number. The mean value $\eta_{ij} d_{ij}$ and variance $\eta_{ij} d_{ij}^2$ of Erlang distribution depend on its two parameters. Since the shape parameter d_{ij} is integer not all the values of mean and variance are simultaneously feasible.

Like in the case of von Mises, Erlang distribution can be represented by a linear combination of orthogonal functions. Since the domain is the positive real line $r \geqslant 0$, the natural basis is given by *Laguerre polynomials* [94]. The Erlang can be expanded in Laguerre polynomial series with coefficients $\{c_{k_r}\}$ as

$$p_{r_{ij}}(r) = \sum_{k_r=0}^{+\infty} c_{k_r}^{ij} \, L_{k_r}(r) \tag{5.9}$$

where $L_{k_r}(r)$ is the Laguerre polynomial of order k_r. The convergence of Laguerre series to a function is guaranteed, when the function satisfies proper conditions [94, Section 4.23]. This condition is satisfied by all the probability density functions. Such polynomials are orthogonal w.r.t. integral inner product with weight function e^{-r}, i.e.

$$\langle L_i, L_j \rangle_{\mathrm{e}^{-r}} = \int_0^{+\infty} \mathrm{e}^{-r} \, L_i(r) \, L_j(r) \, dr = \delta_{ij} \tag{5.10}$$

where δ_{ij} is the Kroenecker delta. Hence, the coefficient $c_{k_r}^{ij}$ is given as

$$\begin{aligned} c_{k_r}^{ij} &= \langle p_{r_{ij}}, L_{k_r} \rangle_{\mathrm{e}^{-r}} && (5.11) \\ &= \sum_{m=0}^{k_r} \binom{k_r}{m} \frac{(-1)^m}{\eta_{ij}^{d_{ij}} (d_{ij}-1)!m!} \int_0^{+\infty} \mathrm{e}^{-\left(1+\frac{1}{\eta_{ij}}\right) r} r^{i+k-1} dr \\ &= \sum_{m=0}^{k_r} (-1)^m \binom{k_r}{m} \binom{m+d_{ij}-1}{m} \frac{\eta_{ij}^m}{(1+\eta_{ij})^{m+d_{ij}}} && (5.12) \end{aligned}$$

The Erlang distribution suffers from some drawbacks. The parameter d_{ij} is equal to the ratio between the square mean value and the variance of the distribution. Hence, when modelling a distribution with constant variance and arbitrary large mean value, the value of d_{ij} may significantly increase leading to potential numeric inaccuracy.

5.3.3 Biased Rayleigh distribution for Pairwise Distance

The *biased Rayleigh* distribution is an alternative model we adopt to model the distribution of pairwise distances. The PDF of biased Rayleigh for $r \geqslant 0$ is

$$p_{r_{ij}}(r) = \frac{1}{\mathrm{K}_{br_{ij}}} \, r \, \exp\left(-\frac{(r-\mu_{ij})^2}{2\sigma_{ij}^2} \right) \tag{5.13}$$

where $\mu_{ij} \geqslant 0$ is the mode and $\sigma_{ij} > 0$ is the width. The mode and the width are parameters setting the shape of the distributions and must not be confused with the mean value and the standard deviation. If $\mu_{ij} = 0$, then Equation (5.13) becomes the PDF of standard Rayleigh distribution. The *unnormalized moment* of order m is defined as

$$
\begin{aligned}
\mathscr{M}_m(\mu,\sigma) &= \int_0^{+\infty} t^{m+1}\,\mathrm{e}^{-\frac{(t-\mu)^2}{2\sigma^2}}\,dt \\
&= \begin{cases} \sigma\,\sqrt{\frac{\pi}{2}}\,\left(1+\operatorname{erf}\left(\frac{\mu}{\sigma\sqrt{2}}\right)\right) & \text{if } m=-1 \\ \sigma\mu\sqrt{\frac{\pi}{2}}\left(1+\operatorname{erf}\left(\frac{\mu}{\sigma\sqrt{2}}\right)\right)+\sigma^2\mathrm{e}^{-\frac{\mu^2}{2\sigma^2}} & \text{if } m=0 \\ \mu\,\mathscr{M}_{m-1}(\mu,\sigma)+m\,\sigma^2\,\mathscr{M}_{m-2}(\mu,\sigma) & \text{otherwise} \end{cases}
\end{aligned}
\tag{5.14}
$$

The value of unnormalized moment is given above in the form of recurrent definition. The moments can be used to find the normalization constant $\mathrm{K}_{br_{ij}}$, the mean value $\mu_{br_{ij}}$ and the variance $\sigma^2_{br_{ij}}$ as

$$
\mathrm{K}_{br_{ij}} = \mathscr{M}_0(\mu_{ij},\sigma_{ij}) \quad \mu_{br_{ij}} = \frac{\mathscr{M}_1(\mu_{ij},\sigma_{ij})}{\mathscr{M}_0(\mu_{ij},\sigma_{ij})} \tag{5.15}
$$

$$
\sigma^2_{br_{ij}} = \frac{\mathscr{M}_2(\mu_{ij},\sigma_{ij})}{\mathscr{M}_0(\mu_{ij},\sigma_{ij})} - \left(\frac{\mathscr{M}_1(\mu_{ij},\sigma_{ij})}{\mathscr{M}_0(\mu_{ij},\sigma_{ij})}\right)^2 \tag{5.16}
$$

The biased Rayleigh PDF can be expanded into a series of Laguerre polynomials as shown in Equation (5.9). The coefficients $c^{ij}_{k_r}$ for this expansion are

$$
\begin{aligned}
c^{ij}_{k_r} &= \int_0^{+\infty} \mathrm{e}^{-r}\,\frac{1}{\mathrm{K}_{br_{ij}}}\,r\,\exp\left(-\frac{(r-\mu_{ij})^2}{2\sigma^2_{ij}}\right) L_j(k_r)dr \\
&= \sum_{m=0}^{k_r}\binom{n}{m}\frac{(-1)^{k_r}\mathrm{e}^{\mu_{ij}-\frac{\sigma^2_{ij}}{2}}}{\mathrm{K}_{br_{ij}}\;m!}\int_0^{+\infty}\mathrm{e}^{-\frac{(r-(\mu_{ij}-\sigma^2_{ij}))^2}{2\sigma^2_{ij}}}\,r^{m+1}dr \\
&= \sum_{m=0}^{k_r}\binom{n}{m}\frac{(-1)^{k_r}\mathrm{e}^{\mu_{ij}-\frac{\sigma^2_{ij}}{2}}}{\mathrm{K}_{br_{ij}}\;m!}\,\mathscr{M}_m(\mu_{ij}-\sigma^2_{ij},\sigma_{ij})
\end{aligned}
\tag{5.17}
$$

5.3.4 Signature Distribution of Point Sets

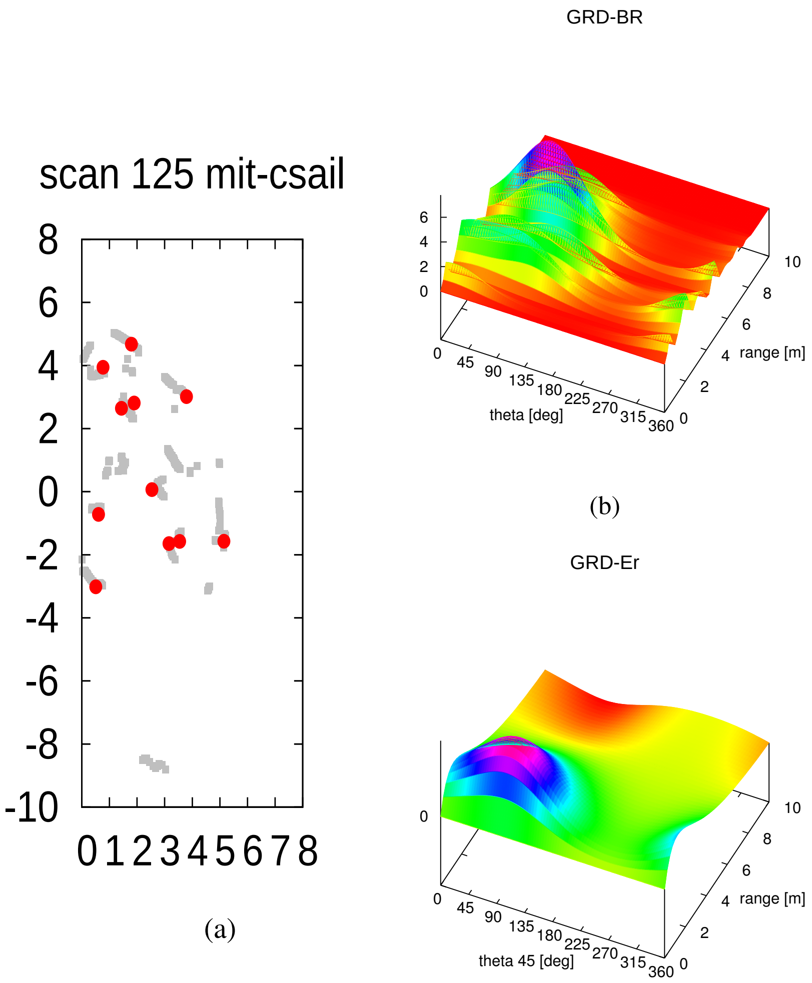
There are all the elements to derive an efficient approximation of the signature distribution in Equation (5.2). The series approximation is based on the approximation of marginals of Equation (5.5) and (5.9). The two series are arrested respectively to order n_θ for random variable θ and n_r for random variable r. Thus, the signature distribution is approximated by $p_{\theta r}(\theta, r) \simeq \mathscr{S}(\theta, r)$ defined as

$$\begin{aligned}\mathscr{S}(\theta,r) &= \frac{1}{N}\sum_{i=1}^{n_p}\sum_{j=i+1}^{n_p}\left(\sum_{k_r=0}^{n_r} c_{k_r}^{ij}\, L_{k_r}(r)\right) \\ &\cdot \left(\sum_{k_\theta=0}^{n_\theta}\left(a_{k_\theta}^{ij}\cos\left(k_\theta\theta\right) + b_{k_\theta}^{ij}\sin\left(k_\theta\theta\right)\right)\right) \\ &= \sum_{k_\theta=0}^{n_\theta}\sum_{k_r=0}^{n_r} \omega_{k_\theta k_r}(\theta, r)\end{aligned} \qquad (5.18)$$

where the terms $\omega_{k_\theta k_r}(\cdot)$ and their parameters are equal to

$$\begin{aligned}&\omega_{k_\theta k_r}(\theta,r) = A_{k_\theta k_r} L_{k_r}(r)\cos\left(k_\theta\theta\right) + B_{k_\theta k_r} L_{k_r}(r)\sin\left(k_\theta\theta\right) \\ &A_{k_\theta k_r} = \sum_{i=1}^{n_p}\sum_{j=i+1}^{n_p}\frac{a_{k_\theta}^{ij} c_{k_r}^{ij}}{N} \quad B_{k_\theta k_r} = \sum_{i=1}^{n_p}\sum_{j=i+1}^{n_p}\frac{b_{k_\theta}^{ij} c_{k_r}^{ij}}{N}\end{aligned} \qquad (5.19)$$

The coefficients $c_{k_r}^{ij}$ in the above equations refer to the Erlang or biased Rayleigh depending on the chosen model. The novel form of Equation (5.18) has the advantage of having a fixed form independently from the number of point pairs N. The complexity of signature computation lies in the coefficients of the above series. The basis functions $L_{k_r}(r)\cos\left(k_\theta\theta\right)$ and $L_{k_r}(r)\sin\left(k_\theta\theta\right)$ are orthogonal. This makes the signature suitable for analytical manipulation as illustrated in the following Section. Figure 5.2 represents the two GRD signatures obtained using respectively biased Rayleigh and Erlang distributions to represent range uncertainty. The signature obtained by Erlang is smoother and less detailed than the one by biased Rayleigh. In general, it is more difficult to shape Erlang kernel PDF according to the desired variance due to discretization and numerical issues of integer exponent d_{ij} in Equation (5.13).



(a)

(b)

(c)

Figure 5.2: Example of GRD signatures computed on a mit-csail dataset: (a) the input scan (grey) with the extracted FLIRT features (red); (b) the GRD-BR signature obtained using biased Rayleigh distribution for ranges (with parameters $(\kappa, \sigma_{ij}) = (1.5, 0.2)$); (c) the GRD-Er signature obtained using Erlang distribution for ranges (with parameters $(\kappa, \sigma_{ij}) = (1.5, 0.2)$). The GRD-Er signature is smoother than GRD-BR one and provides less details.

5.4 A Metric for Signature Distributions

Signature distribution represents the geometric relations of a local map of points. The main goal of signatures is detection of already visited places or, at least, of potential candidate matches. A rotation-invariant metric should be used to compare a given target distribution signature $\mathscr{S}_t$ with a potential matching one $\mathscr{S}_s$. While the pairwise distances are invariant to translation and rotation, the pairwise angles depend on the viewpoint. To achieve invariance, we follow the criterion adopted for GLAROT [57]. The similarity between the signature distributions $\mathscr{S}_t$ and $\mathscr{S}_s$ (apices or pedices s and t are used hereafter) is defined as

$$d(\mathscr{S}_s, \mathscr{S}_t) = \max_{\varphi} \frac{d_\varphi(\mathscr{S}_s, \mathscr{S}_t)}{\sqrt{d_0(\mathscr{S}_s, \mathscr{S}_s)\, d_0(\mathscr{S}_t, \mathscr{S}_t)}} \tag{5.20}$$

where

$$d_\varphi(\mathscr{S}_s, \mathscr{S}_t) = \int_0^{2\pi} d\theta \int_0^{\infty} dr\, \mathrm{e}^{-r}\, \mathscr{S}_s(\theta + \varphi, r)\, \mathscr{S}_t(\theta, r) \tag{5.21}$$

Instead of being a distance, the quantity $d(\cdot)$ is maximum when the two distributions represent the geometric relations of the same point set. The integral in Equation (5.20), which consists of a product of two truncated series, could be difficult to solve. Let $\omega^s_{k_\theta k_r}(\cdot)$ and $\omega^t_{k_\theta k_r}(\cdot)$ be the terms of series $\mathscr{S}_s$ and $\mathscr{S}_t$ according to Equation (5.18), with coefficients respectively $\{A^s_{k_\theta k_r}, B^s_{k_\theta k_r}\}$ and $\{A^t_{k_\theta k_r}, B^t_{k_\theta k_r}\}$. The integral of $\omega^s_{k_\theta k_r}(\cdot)\omega^t_{k'_\theta k'_r}(\cdot)$ is zero if $k_\theta \neq k'_\theta$ or $k_r \neq k'_r$ due to the orthogonality of the basis function. Thus, the only remaining terms are $\omega^s_{k_\theta k_r}(\cdot)\omega^t_{k_\theta k_r}(\cdot)$. In particular, the integral is solved as

$$\begin{aligned}
\zeta_{k_\theta k_r}(\varphi) &= \int_0^{2\pi} \int_0^{\infty} \mathrm{e}^{-r} \omega^s_{k_\theta k_r}(\theta + \varphi, r)\omega^t_{k_\theta k_r}(\theta, r) d\theta dr \\
&= \frac{A^s_{k_\theta k_r} A^t_{k_\theta k_r} + B^s_{k_\theta k_r} B^t_{k_\theta k_r}}{2} \cos(k_\theta \varphi) \\
&\quad + \frac{A^s_{k_\theta k_r} B^t_{k_\theta k_r} - B^s_{k_\theta k_r} A^t_{k_\theta k_r}}{2} \sin(k_\theta \varphi)
\end{aligned} \tag{5.22}$$

After the substitution of Equation (5.22) in the metric expression, we obtain

$$
\begin{aligned}
d_\varphi(\mathscr{S}_s, \mathscr{S}_t) &= \sum_{k_\theta=0}^{n_\theta} \sum_{k_r=0}^{n_r} \zeta_{k_\theta k_r}(\varphi) && (5.23)\\
&= \sum_{k_\theta=0}^{n_\theta} A^{\varphi}_{k_\theta k_r} \cos(k_\theta \varphi) + B^{\varphi}_{k_\theta k_r} \sin(k_\theta \varphi) && (5.24)
\end{aligned}
$$

where the coefficients $\{A^{\varphi}_{k_\theta k_r}, B^{\varphi}_{k_\theta k_r}\}$ are obtained as

$$
\begin{aligned}
A^{\varphi}_{k_\theta k_r} &= \sum_{k_r=0}^{n_r} \frac{A^s_{k_\theta k_r} A^t_{k_\theta k_r} + B^s_{k_\theta k_r} B^t_{k_\theta k_r}}{2} && (5.25)\\
B^{\varphi}_{k_\theta k_r} &= \sum_{k_r=0}^{n_r} \frac{A^s_{k_\theta k_r} B^t_{k_\theta k_r} - B^s_{k_\theta k_r} A^t_{k_\theta k_r}}{2} && (5.26)
\end{aligned}
$$

Figure 5.3 shows the correlation function $d_\varphi(\mathscr{S}_s, \mathscr{S}_t)$ for the keypoint features presented in Figure 5.1. It can be observed that the harmonic component $k_\theta = 1$ dominates over higher frequency components. The estimated angular shift φ, where d_φ is maximum, provides a rather accurate estimation of rotation angle between the two feature maps.

Signatures are used for retrieval of candidates matching local maps. Next the candidates are checked by performing point-to-point association. The *Correspondence Graph* (CG) [57] searches potential correspondences based on the same pairwise geometric relations used by GRD signature. Since CG exploits the consistency of matching points, it does not rely on the estimation of local map reference frames w.r.t. the global reference frame. Figure 5.4 shows the CG association for the previously discussed example.

5.5 Experiments

In this Section our experimental setup and results based on loop closure is presented. The experiments are assessed in four public datasets: *fr079*, *mit-csail*, *intel-lab* and *fr-clinic*. The datasets provided by [81] contain both original scans and corresponding

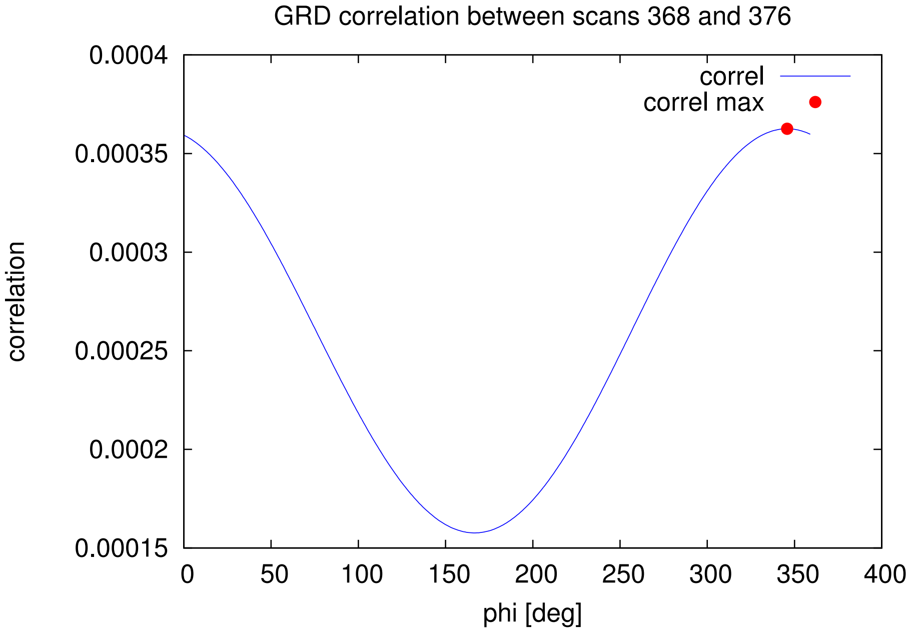


Figure 5.3: Example of GRD signature correlation between the scans 368 and 376 of dataset *intel*. Maximum correlation is achieved with angular shift $\varphi = 345.8$ *deg* (equivalent to -14.2 *deg*) consistent with point rotation (see Figure 5.1).

corrected ground-truth. Results are presented with precision-recall curves for loop closure tests.

The loop closure experiments are evaluated comparing each scan $\mathscr{S}_i$ with the other scans of the dataset $\mathscr{S}_j$ with $j \neq i$. The proposed loop closure algorithm GRD[1], both with Erlang (GRD-Er) and biased Rayleigh (GRD-BR), has been tested in each dataset and compared with two state-of-the-art algorithms, GFP and GLAROT. We evaluate the original GFP algorithm using FLIRT keypoints and descriptors, since signature GFP was originally coupled with the specific keypoint feature. The keypoint features SKIP [95] are used to build the landmark maps encoded by GLAROT and GRD. Signature evaluation requires that loop closure candidates are tested with

---

[1] `https://github.com/dlr1516/grd`.

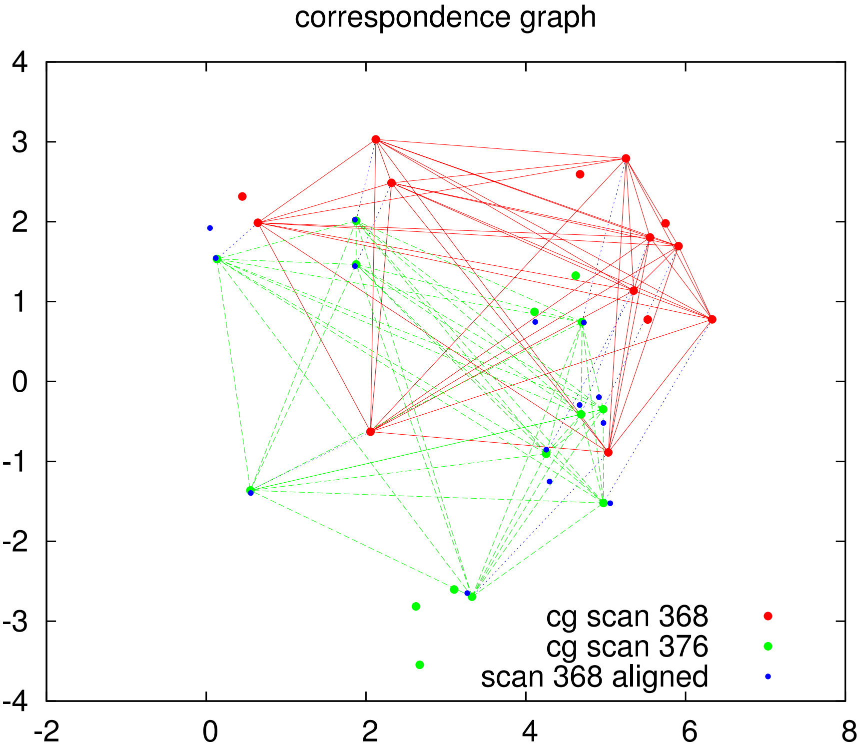


Figure 5.4: Example of point-to-point association using correspondence graph (CG): the keypoint features of scans 368 (red) and 376 (green) in Figure 5.1 with their pairwise relation graphs, and the features 368 aligned to features 376 (blue).

point-to-point association. GFP uses RANSAC algorithm as point-to-point association method according to its original formulation and implementation. The other signatures are evaluated with point-to-point matching method *Correspondence Graph* (CG) [57].

The loop closure evaluation has been performed as follows. For each scan $\mathscr{S}$ of a dataset the scan signatures have been computed and stored. Then, let $\mathscr{S}_R$ be the reference scan, a set $\mathscr{C}_S$ of 20 candidate scans (50 for GFP), whose signatures are closest to the signatures of the reference scan $\mathscr{S}_R$ according to the respective loop closure method, are extracted. The set of closest signatures $\mathscr{C}_S$ is evaluated according to the corresponding distance function and thresholds of each loop closure method. The keypoints of each scan in $\mathscr{C}_S$ are associated with $\mathscr{S}_R$ using the selected point-to-point association algorithm. Then, the robot pose is estimated by solving the orthogonal Procrustes problem on associated point pairs. Finally, the scan selected for loop closure $\bar{\mathscr{S}_i}$ is the scan that, after alignment, has the greatest number of keypoints with a neighbour point in $\mathscr{S}_R$ within the 0.10 m range. The place recognition performance is measured by precision-recall curves. The robot is considered localized if the associated points are at least N_{min}. The precision-recall curves are computed w.r.t. the threshold N_{min}. A localization is considered correct when the position error of the aligned scan is less than 0.50 m and the angular error less than $10°$.

Figures 5.5-5.8 depict the precision-recall curves obtained in datasets *fr079*, *intel-lab*, *mit-csail* and *fr-clinic* with the place recognition algorithms previously discussed.

Table 5.1 shows the computed areas below the aforementioned curves obtained on standard datasets for the considered signatures.

GFP achieves results comparable to the other features in dataset *fr079*, but in the other cases it is dominated by the other approaches. Its performance may be affected by the use of FLIRT features and descriptors used in the local map. Previous works [57, 79] have already observed the potential weakness of local descriptors for LIDAR measurements and the arising limitation of the resulting BoW detection. The curves of GLAROT, GRD-BR and GRD-Er tend to overlap in the case *intel*, but GRD-BR dominates GLAROT across all the other approaches. While all such signatures are based on geometric relations, GRD-BR seems to provide a more accurate representation of cumulated pairwise distances and angles.

Indeed, GLAROT discretizes these parameters into histogram bins whereas GRD-Er exploits Erlang distribution, which achieves smoother and less detailed description than biased Rayleigh, as shown in Figure 5.2. Thus, the proposed GRD signature is able to select potential loop closure candidates with similar or better performance compared to other state-of-the-art methods like GLAROT.

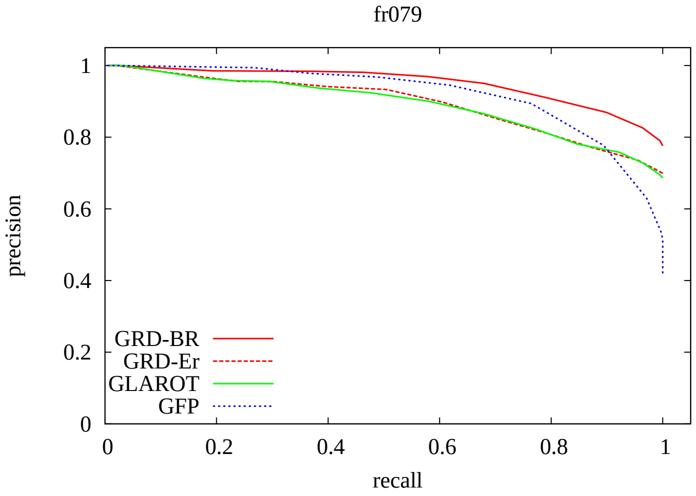


Figure 5.5: Precision-recall curves of loop closure based on GRD-BR, GRD-Er, GLAROT and GFP in datasets *fr079*.

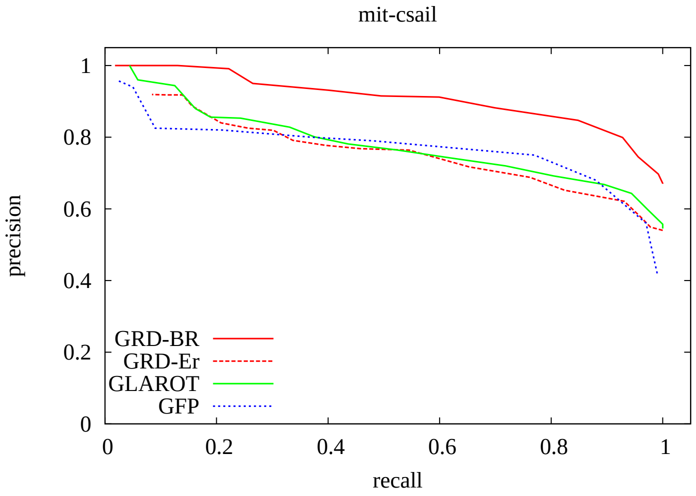


Figure 5.6: Precision-recall curves of loop closure based on GRD-BR, GRD-Er, GLAROT and GFP in datasets *mit-csail*.

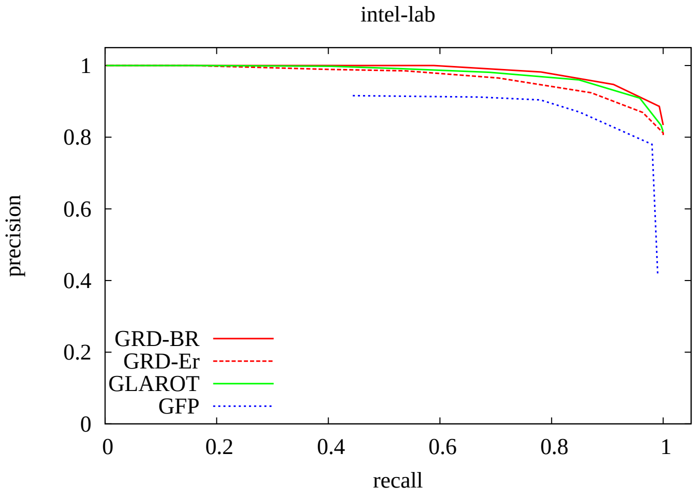


Figure 5.7: Precision-recall curves of loop closure based on GRD-BR, GRD-Er, GLAROT and GFP in datasets *intel-lab*.

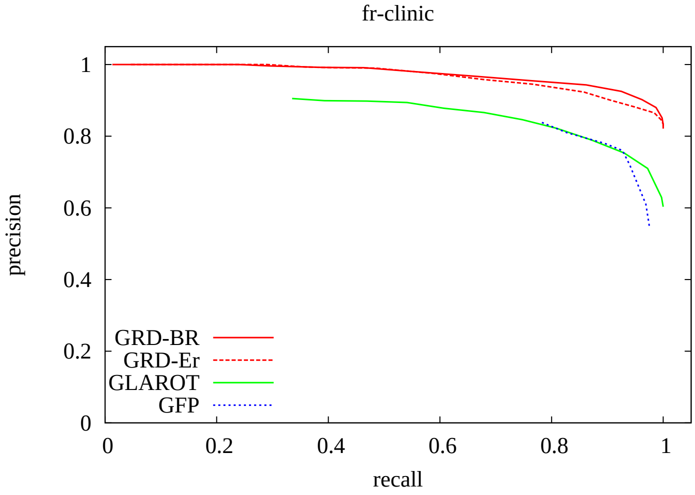


Figure 5.8: Precision-recall curves of loop closure based on GRD-BR, GRD-Er, GLAROT and GFP in datasets *fr-clinic*.

	fr079	*mit-csail*	*intel-lab*	*fr-clinic*
GRD-BR	0.951	0.910	0.986	0.974
GRD-Er	0.895	0.760	0.968	0.968
GLAROT	0.893	0.783	0.979	0.864
GFP	0.921	0.771	0.893	0.819

Table 5.1: Computed areas below the precision-recall curves in Figures 5.5-5.8 with signatures (GRD-BR, GRD-Er, GLAROT and GFP) on standard datasets (*fr079*, *mit-csail*, *intel-lab*, and *fr-clinic*).

operation		avg time [μs]
GFP	construction	2722.0
	pairwise distance	3.7
GLAROT	construction	138.7
	pairwise distance	4.2
GRD-Er	construction	2052.4
	pairwise distance	275.2
GRD-BR	construction	2445.0
	pairwise distance	274.6
CG association		538.6
SKIP detection		97.5

Table 5.2: Average times per scan required for creation and pairwise distance of signatures (GFP, GLAROT, GRD-Er and GRD-BR) and for CG point-to-point association. The times have been computed on datasets *fr079*, *intel-lab*, *mit-csail* and *fr-clinic* with an average number of 13.47 detected features per scan.

The efficiency of the signatures has been empirically assessed by signature matching with datasets *fr079*, *intel-lab*, *mit-csail* and *fr-clinic*. The execution times presented in Table 5.2 are obtained on an Intel i7-3630QM CPU @ 2.40GHz, 8 GB RAM processor running C-based code. In particular, these tests measure the average execution time for the construction of signatures and for the computation of single pairwise distance between two signatures.

A single loop closure query requires the computation of the pairwise distances between the query signature and all the other signatures of the map. The construction time for GRD signatures is about one order of magnitude higher than for GLAROT, but comparable with GFP whose complexity is due to the initial offline computation with BoW dictionary. The complexity of GRD construction is quadratic with the number of point features whereas the pairwise distance computation is computed using guaranteed branch-and-bound optimization with 1 *deg* tolerance on correlation (compared with 22.5 *deg* bin size of GLAROT). Moreover, the computation time could be significantly improved by an optimized implementation and better parameter tuning.

5.6 Discussion

This chapter presented the novel signature GRD for landmark maps, designed to recognize previously visited places for localization and loop closure. The signature corresponds to the probability density function of the pairwise angles and distances among the landmarks of a local map. The distribution models for the two polar coordinates are respectively von Mises for angles and Erlang or biased Rayleigh for distances. The signature GRD has been expanded into series of orthogonal function basis, namely Fourier series and Laguerre polynomials. The coefficients of the series have been derived and used to represent the signature. This representation enables efficient analytic operations with multimodal distribution and, in particular, allows the definition of rotation and translation invariant metric to compare the distribution. The GRD capability to detect visited places in the map has been compared with other state-of-the-art signatures in loop closure experiments.

In future works, we expect to apply GRD in the loop closure component of a localization and mapping system. Moreover, we will investigate the extension of GRD signature to 3D landmark maps and the design of scale-invariant metrics suitable for computer vision applications. Indeed, further investigation and development work is needed before bringing GRD signature-based place recognition into industrial AGVs.

Chapter 6

Conclusion

This dissertation has presented some advanced methodologies to improve positioning and localization performance of industrial AGVs. Based on the methods and techniques reported in this dissertation, the whole warehouse deployment workflow has become considerably faster, simpler and more flexible.

A novel automatic calibration method for Dual Drive and Ackermann AGVs has been presented in Chapter 3. The proposed calibration algorithm estimates both the kinematic parameters and the laser rangefinder parameters by comparing the sensor egomotion and the motion commands. This method has been implemented on industrial vehicles and its real performance have been tested in real warehouses. Since on point accuracies are about $10 - 15$ *mm*, this method returns better results than classic manual calibration. The proposed method outperformed the canonical one even during trials on AGV fleets. The whole procedure takes about 15 *min* for each AGV, reducing calibration times up to 80% and enhancing positioning precision.

Three novel algorithms for AGV localization in unstructured environments have been presented in Chapter 4. The feature method detecting *SKIP* and the registration-based methods *ICP* and *ARS-Grid* have been implemented on industrial AGVs and their performance has been tested in real plants. The aforementioned methods have been compared with the standard artificial landmark localization technique using reflectors. The static map was obtained using a TLS and innovative software processing while measurement ground-truth has been provided by a TS. Localization accuracy is less than 1 *cm*, offering better or comparable performance w.r.t. reflector-based localization. Inaccurate initial guess and laser angular occlusion experiments have shown the robustness of the proposed methods.

A novel signature for place recognition and loop closure with landmark maps, named GRD, has been presented in Chapter 5. This signature encodes pairwise distances and angles between landmarks into a continuous PDF. The effectiveness of the method has been assessed through experiments with standard literature datasets. The lack of industrial experiments highlights the lower technological readiness level of this solution w.r.t. the methods proposed in the earlier chapters, but further development work will surely support industrial improvements.

The combination of the aforementioned techniques can increase positioning precision and improve localization, enhancing the efficiency of the AGV warehouse. Plant deployment and installation times are considerably decreased due to reduction of manual operation and artificial objects installation.

Future works will investigate the possibility of reducing the size of the area required for automatic calibration to better suit the procedure in already existent warehouses. Regarding localization in unstructured environments, further efforts may lead to more robust and industrialized positioning methods, increasing efficiency while reducing errors and installation times.

Bibliography

[1] R. C. Arkin and R. R. Murphy. Autonomous navigation in a manufacturing environment. *IEEE Transactions on Robotics and Automation*, 6(4):445–454, Aug 1990. `doi:10.1109/70.59355`.

[2] L. Schulze and A. Wullner. The approach of automated guided vehicle systems. In *2006 IEEE International Conference on Service Operations and Logistics, and Informatics*, pages 522–527, June 2006. `doi:10.1109/SOLI.2006.328941`.

[3] L. Sabattini, V. Digani, C. Secchi, G. Cotena, D. Ronzoni, M. Foppoli, and F. Oleari. Technological roadmap to boost the introduction of agvs in industrial applications. In *2013 IEEE 9th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 203–208, Sep. 2013. `doi:10.1109/ICCP.2013.6646109`.

[4] L. Sabattini, M. Aikio, P. Beinschob, M. Boehning, E. Cardarelli, V. Digani, A. Krengel, M. Magnani, S. Mandici, F. Oleari, C. Reinke, D. Ronzoni, C. Stimming, R. Varga, A. Vatavu, S. Castells Lopez, C. Fantuzzi, A. Mayra, S. Nedevschi, C. Secchi, and K. Fuerstenberg. The pan-robots project: Advanced automated guided vehicle systems for industrial logistics. *IEEE Robotics Automation Magazine*, 25(1):55–64, March 2018. `doi:10.1109/MRA.2017.2700325`.

[5] R. D'Andrea and P. Wurman. Future challenges of coordinating hundreds of autonomous vehicles in distribution facilities. In *2008 IEEE International Con-*

ference on Technologies for Practical Robot Applications, pages 80–83, Nov 2008. `doi:10.1109/TEPRA.2008.4686677`.

[6] ILIAD - Intra-Logistics with Integrated Automatic Deployment: Safe and Scalable Fleets in Shared Spaces. *on Deliverable 1.2 - Precise localisation and mapping in dynamic environments*, 2018. URL: `https://iliad-project.eu`.

[7] REFILLS - Robotics Enabling Fully-Integrated Logistics Lines for Supermarkets, 2017. URL: `http://www.refills-project.eu`.

[8] P. Wurman, R. D'Andrea, and M. Mountz. Coordinating hundreds of cooperative, autonomous vehicles in warehouses. In *Proceedings of the 19th National Conference on Innovative Applications of Artificial Intelligence - Volume 2*, IAAI'07, page 1752–1759. AAAI Press, 2007.

[9] G. Garibotto, S. Masciangelo, P. Bassino, C. Coelho, A. Pavan, and M. Marson. Industrial exploitation of computer vision in logistic automation: autonomous control of an intelligent forklift truck. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1459–1464, May 1998. `doi:10.1109/ROBOT.1998.677310`.

[10] A. Mäyrä, M. Aikio, K. Ojala, S. Mandici, A. Vatavu, and S. Nedevschi. Fisheye optics for omnidirectional stereo camera performance evaluation for agv applications. 09 2015. `doi:10.1109/ICCP.2015.7312634`.

[11] R. Varga, A. Costea, and S. Nedevschi. Improved autonomous load handling with stereo cameras. In *2015 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 251–256, Sep. 2015. `doi:10.1109/ICCP.2015.7312639`.

[12] T. Stoyanov, R. Mojtahedzadeh, H. Andreasson, and A.-J. Lilienthal. Comparative evaluation of range sensor accuracy for indoor mobile robotics and automated logistics applications. *Robotics and Autonomous Systems*, 2012. `doi:10.1016/j.robot.2012.08.011`.

[13] R. V. Bostelman, T. H. Hong, and R. Madhavan. Towards agv safety and navigation advancement obstacle detection using a tof range camera. In *ICAR '05. Proceedings., 12th International Conference on Advanced Robotics, 2005.*, pages 460–467, July 2005. `doi:10.1109/ICAR.2005.1507450`.

[14] M. Giorgini. *Terrestrial laser scanning as a support to design and deployment of automated warehouses*. PhD thesis, Università degli studi di Parma. Dipartimento di Ingegneria e architettura, 2019. URL: `http://hdl.handle.net/1889/3840`.

[15] P. Beinschob, M. Meyer, C. Reinke, V. Digani, C. Secchi, and L. Sabattini. Semi-automated map creation for fast deployment of AGV fleets in modern logistics. *Robotics and Autonomous Systems*, 87:281 – 295, 2017. `doi:10.1016/j.robot.2016.10.018`.

[16] P. Beinschob and C. Reinke. Advances in 3D data acquisition, mapping and localization in modern large-scale warehouses. In *2014 IEEE 10th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 265–271, Sep. 2014. `doi:10.1109/ICCP.2014.6937007`.

[17] C. Reinke and P. Beinschob. Strategies for contour-based self-localization in large-scale modern warehouses. In *2013 IEEE 9th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 223–227, Sep. 2013. `doi:10.1109/ICCP.2013.6646112`.

[18] F. Kallasi, D. Lodi Rizzini, F. Oleari, M. Magnani, and S. Caselli. A novel calibration method for industrial AGVs. *Robotics and Autonomous Systems (RAS)*, 94:75–88, Aug 2017. DOI 10.1016/j.robot.2017.04.019. `doi:10.1016/j.robot.2017.04.019`.

[19] A. Censi, A. Franchi, L. Marchionni, and G. Oriolo. Simultaneous calibration of odometry and sensor parameters for mobile robots. *TRO*, 29(2), April 2013. `doi:10.1109/TRO.2012.2226380`.

[20] J. Borenstein and L. Feng. Measurement and correction of systematic odometry errors in mobile robots. *IEEE Transactions on Robotics and Automation*, 12(6):869–880, Dec 1996. `doi:10.1109/70.544770`.

[21] T.D. Larsen, M. Bak, N.A. Andersen, and O. Ravn. Location estimation for an autonomously guided vehicle using an augmented Kalman filter to autocalibrate the odometry. In *Proc. Int. Conf. Multisource-Multisens. Inf. Fusion*, pages 1–6, 1998.

[22] A. Martinelli and R. Siegwart. Observability Properties and Optimal Trajectories for On-line Odometry Self-Calibration. In *Proc. of the International Conference on Decision and Control*, pages 3065–3070, 2006.

[23] A. Martinelli, N. Tomatis, and R. Siegwart. Simultaneous localization and odometry self calibration for mobile robot. *Autonomous Robots*, 22(1):75–85, 2007.

[24] D. Caltabiano, G. Muscato, and F. Russo. Localization and self-calibration of a robot for volcano exploration. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, volume 1, pages 586–591 Vol.1, April 2004. `doi:10.1109/ROBOT.2004.1307212`.

[25] G. Antonelli, S. Chiaverini, and G. Fusco. A calibration method for odometry of mobile robots based on the least-squares technique: theory and experimental validation. *IEEE Trans. on Robotics*, 21(5):994–1004, 2005.

[26] G. Antonelli and S. Chiaverini. Linear estimation of the physical odometric parameters for differential-drive mobile robots. *Journal of Autonomous Robots*, 23(1):59–68, 2007. `doi:10.1007/s10514-007-9030-2`.

[27] N. Seegmiller, F. Rogers-Marcovitz, G. Miller, and A. Kelly. Vehicle model identification by integrated prediction error minimization. *Int. Journal of Robotics Research*, 32(8):912–931, 2013.

[28] K. Lee and W. Chung. Calibration of kinematic parameters of a Car-Like Mobile Robot to improve odometry accuracy. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2546–2551, May 2008. `doi:10.1109/ROBOT.2008.4543596`.

[29] M. De Cecco. Self-calibration of AGV inertial-odometric navigation using absolute-reference measurements. In *Proc. IEEE Instrumentation and Measurement Technology Conference (IMTC)*, volume 2, pages 1513–1518 vol.2, 2002. `doi:10.1109/IMTC.2002.1007183`.

[30] P. Goel, S. Roumeliotis, and G.S. Sukhatme. Robust localization using relative and absolute position estimates. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1134–1140, 1999.

[31] J.G. Kang, W.-S. Choi, S.-Y. An, and S.-Y. Oh. Augmented EKF based SLAM method for improving the accuracy of the feature map. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3725–3731, Oct 2010. `doi:10.1109/IROS.2010.5652938`.

[32] E.M. Foxlin. Generalized architecture for simultaneous localization, auto-calibration, and map-building. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, volume 1, pages 527–533, 2002. `doi:10.1109/IRDS.2002.1041444`.

[33] O.D. Faugeras and G. Toscani. Camera calibration for 3-D computer vision. In *Proc. Int. Work. Industrial Applications of Machine Vision and Machine Intelligence*, pages 240–247, 1987.

[34] P. Liang, Y.L. Chang, and S. Hackwood. Adaptive self-calibration of vision-based robot systems. *Systems, Man and Cybernetics, IEEE Trans. on*, 19(4):811–824, 1989.

[35] T. Sasaki and H. Hashimoto. Calibration of laser range finders based on moving object tracking in Intelligent Space. In *Int. Conf. on Networking, Sensing*

and Control (ICNSC), pages 620–625, March 2009. `doi:10.1109/ICNSC.2009.4919349`.

[36] J.H. Song and G.-I. Jee. Kalman filter based on-line calibration of laser scanner for vehicle navigation. In *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*, pages 1437–1441, Oct 2011.

[37] A. Martinelli, D. Scaramuzza, and R. Siegwart. Automatic self-calibration of a vision system during robot motion. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 43–48, May 2006. `doi:10.1109/ROBOT.2006.1641159`.

[38] F.M. Mirzaei and S.I. Roumeliotis. A Kalman Filter-Based Algorithm for IMU-Camera Calibration: Observability Analysis and Performance Evaluation. *IEEE Trans. on Robotics*, 24(5):1143–1156, Oct 2008. `doi:10.1109/TRO.2008.2004486`.

[39] Z. Zhang. A flexible new technique for camera calibration. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, Nov 2000. `doi:10.1109/34.888718`.

[40] R.Y. Tsai and R.K. Lenz. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Trans. on Robotics and Automation*, 5(3):345–358, Jun 1989. `doi:10.1109/70.34770`.

[41] R. Horaud and F. Dornaika. Hand-eye Calibration. *Int. Journal of Robotics Research*, 14(3):195–210, 1995.

[42] B. Schmidt and L. Wang. Automatic work objects calibration via a global–local camera system. *Robotics and Computer-Integrated Manufacturing*, 30(6):678 – 683, 2014. `doi:10.1016/j.rcim.2013.11.004`.

[43] Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2301–2306, 2004.

[44] N. Roy and S. Thrun. Online self-calibration for mobile robots. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, volume 3, pages 2292–2297, 1999. `doi:10.1109/ROBOT.1999.770447`.

[45] A. Kelly. Linearized Error Propagation in Odometry. *Int. Journal of Robotics Research*, 23(2):179–218, Feb. 2004.

[46] D. Cucci and M. Matteucci. A Flexible Framework for Mobile Robot Pose Estimation and Multi-Sensor Self-Calibration. In *Proc. of the Intl. Conf. on Informatics in Control, Automation and Robotics (ICINCO)*, pages 1–8, 2013.

[47] J.P. Underwood, A. Hill, T. Peynot, and S.J. Scheding. Error modeling and calibration of exteroceptive sensors for accurate mapping applications. *Journal of Field Robotics*, 27(1):2–20, 2010. `doi:10.1002/rob.20315`.

[48] J. Brookshire and S. Teller. Automatic calibration of multiple coplanar sensors. In *Robotics: Science and Systems*, pages 33–40. MIT Press, 2011.

[49] J. Brookshire and S. Teller. Extrinsic Calibration from Per-Sensor Egomotion. In *Robotics: Science and Systems*, pages 25–32. MIT Press, 2012.

[50] D. Lodi Rizzini, F. Kallasi, J. Aleotti, F. Oleari, and S. Caselli. Integration of a Stereo Vision System into an Autonomous Underwater Vehicle for Pipe Manipulation Tasks. *Computers and Electrical Engineering (CAEE)*, 58:560–571, feb 2017. DOI 10.1016/j.compeleceng.2016.08.023, EID 2-s2.0-84994797689. `doi:10.1016/j.compeleceng.2016.08.023`.

[51] X. Wu, Y. Zhang, T. Zou, L. Zhao, P. Lou and Z. Yin. Coordinated path tracking of two vision-guided tractors for heavy-duty robotic vehicles. *Robotics and Computer-Integrated Manufacturing*, 53:93 – 107, 2018. `doi:10.1016/j.rcim.2018.03.012`.

[52] G. Vasiljevic, D. Miklic, I. Draganjac, Z. Kovačic, and P. Lista. High-accuracy vehicle localization for autonomous warehousing. *Robotics and*

Computer-Integrated Manufacturing, 42:1 – 16, 2016. `doi:10.1016/j.rcim.2016.05.001`.

[53] H. Martìnez-Barberà and D. Herrero-Pèrez. Autonomous navigation of an automated guided vehicle in industrial environments. *Robotics and Computer-Integrated Manufacturing*, 26(4):296 – 311, 2010. `doi:10.1016/j.rcim.2009.10.003`.

[54] P.J. McKerrow and D. Ratner. Calibrating a 4-wheel mobile robot. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 859–864, 2002.

[55] D. Hess, F. Kuenemund, and C. Roehrig. Simultaneous calibration of odometry and external sensors of omnidirectional automated guided vehicles (agvs). In *International Symposium on Robotics (ISR)*, pages 480–487, 2016.

[56] J. Maye, H. Sommer, G. Agamennoni, R. Siegwart, and P. Furgale. Online self-calibration for robotic systems. *Int. Journal of Robotics Research*, 35(4):357–380, 2016. `doi:10.1177/0278364915596232`.

[57] F. Kallasi and D. Lodi Rizzini. Efficient Loop Closure based on FALKO LIDAR Features for Online Robot Localization and Mapping. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1206–1213, 2016.

[58] A. Kelly and N. Seegmiller. A vector algebra formulation of mobile robot velocity kinematics. In *Field and Service Robotics*, pages 613–627. Springer, 2014.

[59] Gilbert Strang. *Linear algebra and its applications.* Thomson, Brooks/Cole, Belmont, CA, 2006. URL: `http://www.amazon.com/Linear-Algebra-Its-Applications-Edition/dp/0030105676`.

[60] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[61] F. Galasso, D. Lodi Rizzini, F. Oleari, and S. Caselli. Efficient calibration of four wheel industrial AGVs. *Robotics and Computer-Integrated Manufacturing (RCIM)*, 57:116–128, jun 2019. `doi:10.1016/j.rcim.2018.11.005`.

[62] M.R. Walter, S. Karaman, E. Frazzoli, and S. Teller. Closed-loop pallet manipulation in unstructured environments. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 5119–5126, 2010. `doi:10.1109/IROS.2010.5652377`.

[63] M. Seelinger and J.-D. Yoder. Automatic visual guidance of a forklift engaging a pallet. *Robotics and Autonomous Systems*, 54(12):1026 – 1038, 2006. `doi:https://doi.org/10.1016/j.robot.2005.10.009`.

[64] T. Stoyanov, R. Mojtahedzadeh, H. Andreasson, and A.J. Lilienthal. Comparative evaluation of range sensor accuracy for indoor mobile robotics and automated logistics applications. *Robotics and Autonomous Systems*, 61(10):1094 – 1105, 2013. `doi:10.1016/j.robot.2012.08.011`.

[65] H. Andreasson, A. Bouguerra, B. Åstrand, and T. Rögnvaldsson. *Gold-Fish SLAM: An Application of SLAM to Localize AGVs*, pages 585–598. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. `doi:10.1007/978-3-642-40686-7_39`.

[66] H. Martinez-Barbera and D. Herrero-Perez. Autonomous navigation of an automated guided vehicle in industrial environments. *Robotics and Computer-Integrated Manufacturing*, 26(4):296 – 311, 2010. `doi:10.1016/j.rcim.2009.10.003`.

[67] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, Cambridge, MA, 2005.

[68] M. Giorgini, J. Aleotti, and R. Monica. Floorplan generation of indoor environments from large-scale terrestrial laser scanner data. *IEEE Geoscience and Remote Sensing Letters*, 16(5):796–800, May 2019. `doi:10.1109/LGRS.2018.2880042`.

[69] M. Giorgini, F. Barbieri, and J. Aleotti. Ground segmentation from large-scale terrestrial laser scanner data of industrial environments. *IEEE Robotics and Automation Letters*, 2(4):1948–1955, Oct 2017. `doi:10.1109/LRA.2017.2715378`.

[70] D. Holz, A.E. Ichim, F. Tombari, R.B. Rusu, and S. Behnke. Registration with the point cloud library: A modular framework for aligning in 3-d. *IEEE Robotics Automation Magazine*, 22(4):110–124, Dec 2015. `doi:10.1109/MRA.2015.2432331`.

[71] D. Lodi Rizzini. Angular Radon Spectrum for Rotation Estimation. *Pattern Recognition*, 84:182–196, dec 2018.

[72] L.J. Latecki and R. Lakamper. Shape similarity measure based on correspondence of visual parts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(10):1185–1190, Oct 2000. `doi:10.1109/34.879802`.

[73] P.J. Besl and H.D. Mckay. A method for registration of 3-d shapes. *IEEE Trans. Pat. Anal. Mach. Intel*, 14(2):239–256, 1992.

[74] H. Moravec and A. Elfes. High resolution maps from wide angle sonar. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, volume 2, pages 116–121, 1985. `doi:10.1109/ROBOT.1985.1087316`.

[75] W. Hess, D. Kohler, H. Rapp, and D. Andor. Real-Time Loop Closure in 2D LIDAR SLAM. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 1271–1278, 2016. `doi:10.1109/ICRA.2016.7487258`.

[76] E. Olson. Real-time correlative scan matching. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4387–4393, Kobe, Japan, June 2009. IEEE.

[77] G. D. Tipaldi, L. Spinello, and K. O. Arras. Geometrical flirt phrases for large scale place recognition in 2d range data. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2693–2698, 2013.

[78] M. Himstedt, J. Frost, S. Hellbach, H.-J. Boehme, and E. Maehle. Large scale place recognition in 2D lidar scans using geometrical landmark relations. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 5030–5035, 2014.

[79] F. Kallasi, D. Lodi Rizzini, and S. Caselli. Fast keypoint features from laser scanner for robot localization and mapping. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):176–183, jan 2016. DOI 10.1109/LRA.2016.2517210. `doi:10.1109/LRA.2016.2517210`.

[80] J. Deray, J. Solà, and J. Andrade-Cetto. Word ordering and document adjacency for large loop closure detection in 2-d laser maps. *IEEE Robotics and Automation Letters (RA-L)*, 2(3):1532–1539, 2017. `doi:10.1109/LRA.2017.2657796`.

[81] G. D. Tipaldi and K. O. Arras. Flirt-interest regions for 2d range data. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 3616–3622, 2010.

[82] J.-S. Gutmann and K. Konolige. Incremental Mapping of Large Cyclic Environments. In *Proc. of the IEEE Int. Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, pages 318–325, 1999.

[83] M. Himstedt and E. Maehle. Geometry matters: Place recognition in 2D range scans using Geometrical Surface Relations. In *Proc. of the European Conference on Mobile Robots (ECMR)*, pages 1–6, 2015. `doi:10.1109/ECMR.2015.7324185`.

[84] T. Bailey, E.M. Nebot, J.K. Rosenblatt, and H.F. Durrant-Whyte. Data association for mobile robot navigation: a graph theoretic approach. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2512–2517, 2000.

[85] P. San Segundo and D. Rodriguez-Losada. Robust global feature based data association with a sparse bit optimized maximum clique algorithm. 29(5):1332–1339, Oct 2013. `doi:10.1109/TRO.2013.2264869`.

[86] J. Neira and J.D. Tardós. Data Association in Stochastic Mapping Using the Joint Compatibility Test. *IEEE Trans. on Robotics*, 17(6):890–897, 2001.

[87] F. Lu and E. Milios. Globally Consistent Range Scan Alignment for Environment Mapping. *Journal of Autonomous Robots*, 4:333–349, 1997.

[88] E. Mendes, P. Koch, and S. Lacroix. ICP-based pose-graph SLAM. In *IEEE Int. Sym. on Safety, Security, and Rescue Robotics (SSRR)*, pages 195–200, 2016. `doi:10.1109/SSRR.2016.7784298`.

[89] D. Lodi Rizzini. Place Recognition of 3D Landmarks based on Geometric Relations. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 648–654, 2017. `doi:10.1109/IROS.2017.8202220`.

[90] Y. Tazaki, Y. Miyauchi, and Y. Yokokohji. Loop detection of outdoor environment using proximity points of 3d pointcloud. In *2017 IEEE/SICE International Symposium on System Integration (SII)*, pages 411–416, Dec 2017. `doi:10.1109/SII.2017.8279247`.

[91] G. Kim and A. Kim. Scan context: Egocentric spatial descriptor for place recognition within 3d point cloud map. In *In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018.

[92] J. Aleotti, D. Lodi Rizzini, R. Monica, and S. Caselli. Global Registration of Mid-Range 3D Observations and Short Range Next Best Views. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3668–3675, 2014. `doi:10.1109/IROS.2014.6943077`.

[93] M. Abramowitz and I.A. Stegun. *Handbook of Mathematical Functions*. Dover Publications, 1965.

[94] N.N. Lebedev. *Special Functions and Their Applications*. Prentice-Hall Inc., Englewood Cliffs, N.J., 1965.

[95] D. Lodi Rizzini, F. Galasso, and S. Caselli. Safe Feature-based Navigation for Industrial AGVs. In *Proc. of IROS Workshop on Robotics for logistics in warehouses and environments shared with humans*, 2018.