J Intell Robot Syst (2012) 68:121–145
DOI 10.1007/s10846-012-9673-x

How the Location of the Range Sensor Affects EKF-based Localization

Fulvio Mastrogiovanni · Antonio Sgorbissa · Renato Zaccaria

Received: 9 November 2011 / Accepted: 28 March 2012 / Published online: 13 April 2012

Abstract The article proposes a solution to map-based self-localization for an autonomous robot operating in cluttered and crowded environments. To detect features for localization, 2D laser range-finders traditionally scan a plane parallel to the floor. This work hypothesizes the existence of a “low frequency cross-section” of the 3D Workspace where cluttered and dynamic environments become “more regular” and “less dynamic”. The contribution of the article is twofold. First, an “unevenness index” U is introduced to quantitatively measure the complexity of the environment as it would be perceived if the laser range-finder were located at different heights from the floor. The article shows that, by choosing the laser scanning plane to statistically minimize U (in most cases, above the heads of people), it is possible to deal more efficiently with non-linearities in the measurement model, moving objects and occluded features. Second, it is demonstrated that, when adopting an extended Kalman filter for position tracking (a very common and widely used technique in real-world scenarios), the a posteriori covariance of the estimated robot pose converges faster, on average, when U is lower, which leads to better localization performance. Experimental results show hours of continuous robot operation in real-world, cluttered and crowded environments.

Keywords Map-based localization · Extended Kalman filter · Human environment

1 Introduction

When considering complex navigation tasks in real-world applications (e.g., for object transportation and delivery or automated surveillance), self-localization is still a central issue. Since incremental dead-reckoning methods suffer from cumulative errors increasing over time [2, 9, 15, 33, 43, 51, 52, 60, 68], the usual approach is to periodically compare observations with an internal representation of the environment, and to consequently estimate the robot pose. This process is often grounded in the Bayesian estimation framework. Among the most common approaches, Kalman filter-based localization assumes that a uni-modal probability distribution is adequate to maintain the pose estimate [15, 27, 34, 49, 62, 63], thereby being a simple and effective solution to position

F. Mastrogiovanni (✉) · A. Sgorbissa · R. Zaccaria
University of Genova, Genova, Italy
e-mail: fulvio.mastrogiovanni@unige.it

A. Sgorbissa
e-mail: antonio.sgorbissa@unige.it

R. Zaccaria
e-mail: renato.zaccaria@unige.it

tracking when an initial estimate of the robot pose is available. Approaches based on Markov [22–24, 53, 65] and particle filters [25, 32, 46, 50, 64, 66] are able to represent multi-modal probability distributions over the robot poses, thereby allowing for performing global localization. Finally, within the same statistical framework, each new observation can be used to update the map itself, thus simultaneously estimating the robot pose as well as the configuration of surrounding features, a process referred to as Simultaneous Localization And Mapping (SLAM) [21, 26, 32, 46, 50, 67].

Whichever approach is adopted, the localization system requires a model of the noise affecting both sensors [1, 34, 54, 55] and the process to be estimated. Furthermore, the problem of feature association (a prerequisite for any configuration update) dramatically arises when robots are employed in environments where people perform their everyday activities. Offices, hospitals and storage areas are messy and cluttered. Worse than that, humans have the "bad habit" of moving around, carrying objects—which were supposed to be features up to a moment before—from one place to another. In this situation, feature association can be a hard task, either when the map is perfectly known or when it must be *learnt* through observations. In particular, the difficulty relies in finding a subset of features that are "stable" enough to be repeatedly detected and associated, thereby allowing for a reliable correction of the robot pose [36, 37, 62, 64].

Let us consider a very common localization architecture, where a laser range-finder is used to collect observations (e.g., line features), feature association is performed with respect to a known map of the environment and an extended Kalman filter is employed to maintain an updated estimate of the robot pose.[1] Observations are collected (in the form of extracted line features) by scanning a 2D plane parallel to the floor. When a sufficient number of extracted features is associated with corresponding features in the map, the Kalman filter is used to correct the robot position and orientation. According to this localization scheme, it seems convenient to search for a 2D cross-section of the robot Workspace (intuitively, *above* human heads!) where feature association can be considered extremely reliable, even in environments otherwise cluttered and dynamic, and therefore localization performance is expected to improve. Such a 2D cross-section is referred to as "Low Frequency Cross-Section" (LFCS), where the "Low Frequency" attribute refers both to the spatial and to the temporal aspects. As a consequence, three research hypotheses can be made.

Hypothesis 1 If a LFCS exists, moving objects are expected to interfere as less as possible with feature detection, therefore increasing the likelihood of reliable feature association. Furthermore, the resulting map should comprise a smaller number of features, thereby reducing both memory requirements and the computational complexity of localization algorithms. It could be argued that a reduced number of features is a drawback, since the presence of more features could help in disambiguating among different areas of the environment (i.e., for global and topological localization). However, global localization is infeasible with approaches based on Kalman filtering, because the latter assumes the pose error to be modelled as a uni-modal Gaussian distribution.

Hypothesis 2 When range measurements contribute to a smaller number of best fitting line features, it is possible to consider straight infinite lines as features to be extracted from range data (instead of "corners", "finite length segments", or "line intersections"). As a consequence, detected features can be expressed as a linear function of the robot state and the noise affecting observations can be reasonably modelled as Additive White Gaussian (i.e., the system is very close to meet the ideal requirements of the linear Kalman filter).

Hypothesis 3 When range measurements contribute to a smaller set of "longer" and "more reliable" features, the a posteriori covariance is

[1] Many considerations made throughout the article are still valid for other architectural choices and for SLAM as well.

expected to converge faster (on average), therefore increasing localization accuracy.[2]

The article is organized as follows. Section 2 describes how approaches in the literature deal with the problem of dynamic environments. Section 3 validates our intuitions about the LFCS, and introduces the "unevenness index" as a parameter to quantitatively mark the world as perceived by the laser range-finder. In Section 4 the system architecture is described: in particular, the focus is on the effects of the LFCS approach in terms of robot design, linearity of the measurement model and state observability. In Section 5, the effects on the estimated a posteriori covariance are shown. Section 6 discusses experimental results concerning both unevenness measurements, with hundreds of data sets collected in different human populated areas, and the localization process, performed both in simulation and with different robots (a TRC Labmate mobile platform, and our self-designed robots Staffetta, ANSER and Merry Porter). Conclusions follow.

2 Related Work

In localization approaches based on Kalman filtering the robot state (i.e., the pose) is modelled as a discrete-time process governed by the stochastic difference equations:

$$\begin{aligned} \chi_k &= f(\chi_{k-1}, u_k, w_{k-1}), \\ z_k &= h(\chi_k, v_k), \end{aligned} \tag{1}$$

where $\chi = \left(x, y, \theta\right)^T$ is the robot state, u is the driving function, w is the process noise, z is the current observation, and v is the measurement noise. Features extracted from sensory data (e.g., range measurements provided by a laser range-finder) are compared with an a priori available map of the environment to correct the prediction of the robot pose at time step k.

Unfortunately, the ideal conditions for using the Kalman filter are almost never met. It is known that convergence and optimality characteristics of Kalman filtering require the process f and the measurement model h to be linear, and the process and measurement noises w and v to be zero-mean Additive White Gaussian (AWGN). The following discussion focuses on the measurement model and noise, since the process depends on the electromechanical characteristics of the mobile platform, thereby being independent on the sensor and the approach adopted for self-localization.

The observation vector z contains the parameters characterizing each feature extracted from raw range measurements. In the very common case in which z describes the position—relative to the robot reference frame—of features like "corners", "finite length segments", or "line intersections", it is straightforward to verify that the measurement model h involves trigonometric functions and therefore it can not be expressed as a linear function of the state. As a matter of fact, even if it can be reasonably assumed that the radial and angular noise on each range measurement returned by the laser sensor are AWGN [1, 34, 54, 55], the spectral shape of the noise affecting z depends on how single range measurements are combined together to extract features. As a consequence, non-linearities can emerge. The linearisation of the measurement model and the presence of non-AWGN components in the measurement noise are two factors deserving the greatest attention for they are possible causes of performance degradation. Their undesirable effects on state estimation in general [44] and map-based self-localization in particular [35], such as the tendency of seriously underestimating the a posteriori covariance, have been widely pointed out. A method to avoid a non-linear measurement model has been proposed in [35], where the map is made up of features that are invariant with respect to the robot pose.

Localization performance is also affected by the presence in the map of *dynamically changing* and *unmodelled* features. Dynamically changing features are the obvious consequence of a

[2] Notice that, throughout the article, the term "longer" (and its opposite "shorter") is always hyphenated, when referring to lines extracted by raw range data. Since lines obviously have an infinite length, the term "longer" should always be meant as "originated from a larger number of collinear, consecutive range measurements".

crowded environment. They include people moving around the robot, objects moved from one place to another or opened and closed doors. Unmodelled features are the consequence of the intrinsic difficulty of obtaining an accurate map, which is related to the difficulty of selecting stable features: should chairs, tables, waste baskets considered stable? What about computers and displays? Both "noise sources" are particularly critical since they are the major cause for unreliable feature association, a phase that logically precedes pose estimation by providing the correct association between each observed feature and its correspondence in the map. When considering real-world environments, it is then fundamental to filter away the sensor readings that do not have correspondences in the world model. It is worth noting that filtering techniques play a primary role both for map-based localization and SLAM: even if, in principle, SLAM deals with dynamically changing and unmodelled features by continuously updating the map, it still requires a significant amount of features to be repeatedly observed and correctly associated to preserve map consistency [36, 37, 62, 64].

In [22–24] two different filtering techniques are proposed. The former estimates the state entropy value before and after updating the state itself with a new observation. Only those observations reducing this entropy indicator are used. The latter rejects observations mismatching a measure related to the expected distance using a threshold mechanism. In [13, 61] a sample-based joint probabilistic data association approach (based on Particle filters) is used to estimate the position of people surrounding the robot and then to perform map building. An algorithm capable of discriminating between static and dynamic parts of the environment is described in [70, 71]: two separate maps are used and separately updated. In [18] a framework is proposed based on the Rao-Blackwellized genetic algorithmic filter, which uses the negative readings returned by sensors during the SLAM process to identify moving objects in a static map. A similar approach is also used in [39, 45] to address SLAM, which is combined—respectively—with a Kalman filter and a recursive conditional Particle filter to track people close to the robot. In [17] a new SLAM approach is proposed based on topological maps, where the concept of variant and invariant nodes is introduced to deal with a semi-permanent dynamics induced by door opening and closing. The work discussed in [7] considers the specific problem of managing poor-reflecting surfaces identified by erroneous maximum-range readings. The article presents a novel approach explicitly taking reflection properties of surfaces into account, and considering the expectation of valid range measurements to achieve significantly improved localization results. A motion detector based on statistical assumptions has been used in [72] to manage the detection and representation of dynamic objects.

In a "smart environments" perspective, the work discussed in [74] assumes that objects are provided with RFID tags with unique ubiquitous identification codes, which allows the system to detect objects occasionally located in different places, such as mobile file cabinets, chairs, and doors. The work described in [28] proposes a network of laser range-finders located in the environment to detect robots and dynamic objects as well as to recognize people trajectories. A slightly different approach is pursued by [12], which adopts both on-board and distributed laser range-finders. Similar problems have been considered in outdoor localization scenarios: a real-time algorithm for SLAM is presented in [69], which performs detection and tracking of moving objects in dynamic outdoor environments from a moving vehicle equipped with a laser scanner, whereas the work in [57] focuses on data association algorithms to perform outdoor loop closure in presence of moving objects.

SLAM techniques need to deal with the additional problem of the huge *amount of features* typically present in real-world scenarios. As a matter of fact, optimal solutions to the SLAM problem usually exhibit a high computational complexity, which is mostly due to the manipulation of matrices quadratically scaling with the size of the state vector and hence with the number of features detected in the environment and stored into the map. A lot of effort has been devoted to design simplified, possibly sub-optimal algorithms able to cope with a wide, complex environment in real-time [21, 32, 46, 50, 67].

In this perspective, Guivant and Nebot [29] shows a real-time implementation of the SLAM algorithm obtained using a modified version of the extended Kalman filter. The key idea is to consider mutually distant features as uncorrelated, thereby updating only a subspace of the entire state vector corresponding to a local map of the environment. The work in [16] pushes this idea further by providing a feature selection strategy reducing the computational load of the full SLAM update. To a similar aim, a 2-layer approach is adopted in [30], where local maps are continuously updated and the full SLAM update is performed only in particular conditions. The resulting algorithm is thus characterized by the computational complexity $O(n \times n_l)$, where n is the total number of landmarks and n_l is the number of landmarks locally perceived. The authors of [4] present a set of robust and efficient algorithms with $O(n)$ cost for object detection with laser ranger-finders, mobile robot pose estimation, and SLAM improved implementation. Object detection is mainly based on a novel multiple line fitting method, which relies on the evidence that walls in structured environments usually constitute regular constrained angles. To further improve efficiency, the work in [10] presents an algorithm for matching laser-built maps using histogram correlations, a representation summarizing salient features in a map. Hierarchical approaches to SLAM have been proposed as well, integrating topological and metric maps [3, 5, 8, 14, 20, 40, 42, 56], as well as improving the efficiency in data association [47, 61, 67].

Currently, the literature contains many successful examples of localization and SLAM techniques used to maintain 2D and 3D models of the Workspace in real-time: current state-of-the-art algorithms allow for the consistent maintaining of a high number of landmarks [31, 46, 67]. With the availability of efficient algorithms to perform loop-closure in real-time [38], 2D laser-based SLAM is considered a solved problem from the theoretical standpoint. However, the computational efficiency of SLAM approaches is still an important issue.

3 The Low Frequency Cross-section

The key insight related to the "Low Frequency Cross-Section" is that the complex nature of structured human populated environments depends on elevation from the floor: the environment is more variable (both spatially and temporally) under a given height (in easy reaching distance) rather than above human heads. This is due to the presence of people as well as of objects (i.e., computers books and chairs) and furniture that are moved from one location to another, according to contingent needs. This is not surprising: since human beings tend to minimize physical efforts when performing manual tasks, everyday objects are usually at reaching distance, i.e., conveniently for arms and hands. In this perspective, it is easy to imagine that features for robot localization are likely to be randomly occluded depending on everyday human activity, with obvious consequences for the "stability" of the map representation over time, when the laser scanning plane intersects the typical human personal space. Furthermore, even if we assume no human activity (e.g., during night time operation), experience suggests that the environment remains "more regular" and "simpler" over a given height, as the result of previous human activity. This fact definitely impacts the feature extraction process, for instance when computing the line segments best fitting range data: if executed on range data taken over a given height, feature extraction produces on average a smaller number of line segments, each one being—in a sense—"longer" and more reliable (since it is originated from a larger number of range points).

The term "Low Frequency Cross-Section" is adopted to capture these concepts. Figure 1 shows a lab environment with scan profiles taken at different heights, namely *bottom view* (10 cm from the floor), *middle view* (90 cm) and *top view* (200 cm). Intuitively, by opportunely raising the laser scanning plane up to the *top view*, the effect of dynamically changing and not modelled features is expected to become negligible, until reaching the limit situation in which feature association is a trivial task even when the Workspace is crowded with people. Furthermore, a reduced

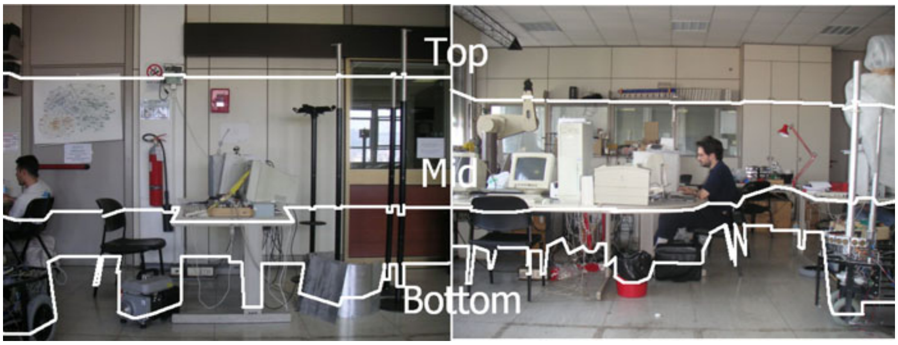


Fig. 1 Two examples of low-frequency cross-secton in a office-like environment: *bottom*, *mid* and *top* views

number of features is expected to substantially reduce the complexity of maintaining a coherent world model, which is especially significant when performing SLAM-like tasks.

The idea of a "freely sensing area" is not new. A well-known approach is described in [25, 65], where the authors deal with the localization problem by pointing a TV camera upward, with the purpose of observing ceiling lights. Another approach is presented in [73], where a 3D laser range-finder is used to observe the ceiling, detecting and matching such structures as beams, columns, air conditioning and lightning installation against a world model containing line and point features. The LFCS approach shares a number of similarities with these and other examples of "opportunistic design" in the literature. However, its peculiarity is straightforward: LFCS is a *generic* method for laser-based feature extraction, which introduces a technique to select the most promising scanning plane (i.e., the *top view*) in virtually any indoor environment.

In order to provide a quantitative evidence of these intuitive concepts, an indicator is introduced to compare different range-finder poses and to quantify the tendency of range data to be collinear (i.e., their spatial frequency) and to be stationary when the robot is not moving (i.e., their temporal frequency). In principle, different indicators borrowed from the field of material roughness and cartography, such as Average Roughness R_a, Root-Mean-Square (RMS) Roughness R_q, Average Absolute Slope Δa, or RMS Average Slope Δq, could be used. Experimental results show that all the indicators, when used to determine the "roughness" of a set of range data, provide similar results in most cases. Among them, we select the "unevenness index": unevenness measures and the fractal degree are commonly used to measure the length and complexity of shore profiles, and can be reasonably applied to quantify the regularity of a given laser scan. The unevenness index U is computed as:

$$U = \frac{P}{C}. \tag{2}$$

In Eq. 2, P is the length of the polyline connecting all the raw scan points, whereas C—commonly referred to as the "chord" of the polyline—is the length of the segment located between the first and the last considered scan points. Apparently, the behaviour of U is adequate to capture the concepts expressed so far.

U is lower bounded This comes from the fact that the link between the start and the final points of a polyline is always shorter than the polyline itself, therefore U is always greater than 1.

U is a measure of the discontinuity between contiguous scan points High U values correspond to a high disparity between contiguous range measurements, whereas low U values are detected if scan points are almost collinear.

Unfortunately, the unevenness index U (as well as all the other possible indicators) exhibits undesired behaviours.

U increases with the depth of the environment That is, long corridors are mainly characterized by high values of U, while rooms yield lower values.

U is not upper bounded When the length of the chord tends to 0, then U tends to reach *infinite*. This case is realistic when full 360 deg scans are considered, e.g., with two different range-finders pointing in opposite directions.

These drawbacks can be mitigated by dividing a full scan into disjoint subsets of range measurements, and considering each polyline separately. As a matter of fact, a similar approach is adopted in cartography, where coastal profiles are divided into "mostly convex" and "mostly concave" segments, which allows to compute the fractal degree for each segment. Unfortunately—and in contrast with the assumptions made for a "good" indicator—the cartographic approach seems to ultimately depend on the algorithm adopted for obtaining disjoint subsets (i.e., scan segmentation), thereby lacking generality.

The problem can be overcome by considering the ratio between Us taken at different heights from the floor, instead of absolute measurements. Let us consider Fig. 2 on the left hand side, which shows simulated range measurements taken at different heights. The "shape" of the room is roughly the same when considering different scans, since it mainly depends on wall profiles. However, local differences are present, due to the presence of occluding objects. In particular, when walls are occluded, the resulting polyline P_2 is usually longer than P_1 (for elementary geometric considerations), and U_2 is consequently greater than U_1 if $C_1 \simeq C_2$. This condition holds in most cases, since the two chords C_1 and C_2 are expected to depend on the "shape" of the whole environment, rather than on local differences. To stress this property, it is arbitrarily assumed that $C_1 = C_2$ even when they are different, which allows for defining the "unevenness ratio" $\Upsilon_{i,j}$ between U_i and U_j as:

$$\Upsilon_{i,j} = \frac{U_i}{U_j} = \frac{P_i/C_i}{P_j/C_j} \approx \frac{P_i}{P_j}. \tag{3}$$

The ratio $\Upsilon_{i,j}$ in Eq. 3 depends only on local differences between the two polylines P_i and P_j, and it is therefore independent from the "shape" of the environment. $\Upsilon_{i,j}$ can be efficiently used to compare different laser scanning planes, thereby allowing for confirming the intuitions related to the LFCS. All the collected data (see Section 6.1) confirm that, when scanning a plane close to the floor level (e.g., for $U = U_{\text{bottom}}$), U is bigger, whereas it significantly decreases as we move the laser to a higher elevation (e.g., $U = U_{\text{mid}}$ or $U = U_{\text{top}}$), where transitions between neighbouring scan points are generally smoother (e.g., in most environments $\Upsilon_{\text{top,bottom}} \ll 1$ and $\Upsilon_{\text{top,mid}} \ll 1$). Moreover, in a dynamic environment, experiments validate the idea that the time variance associated with U is higher when scans are taken at a lower height.

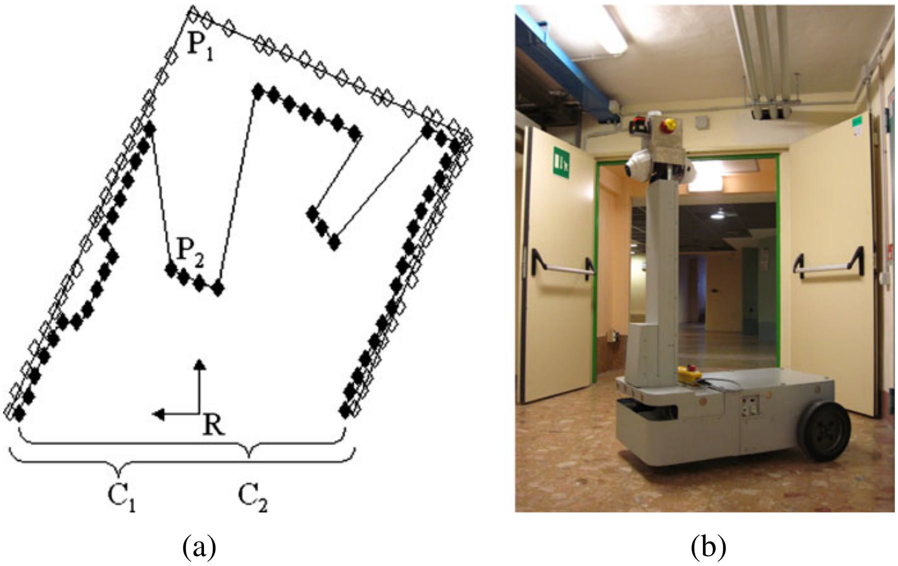


Fig. 2 (**a**) Superimposition of two different polylines taken at different heights from the floor; (**b**) The Merry Porter robot is provided with two different range-finders for localization and obstacle avoidance

4 System Architecture

The adoption of the LFCS approach can have significant consequences on different robot HW/SW architectural choices. This Section shows first how both HW design and the robot SW architecture are affected by the existence of a LFCS. Afterwards, the resulting map-based localization subsystem is discussed in detail, with a specific emphasis on: (1) feature extraction and matching and (2) state estimation through Kalman filtering.

4.1 Hardware Design and Software Architecture

When it is argued that the LFCS approach suggests to mount the laser range-finder at a certain height from the floor, both robot HW and SW aspects are involved. Obviously enough, a laser range-finder located "above the heads of people" can not detect obstacles and people in the nearby. However, best design practices suggest to decouple localization from the obstacle detection and avoidance, in order to minimize the chances of wrong data association (see [11] and the references therein).

With respect to HW issues, decoupling means using two distinct range sensors, the first (for localization) in the LFCS, the second (for obstacle avoidance) outside it. Figure 2 on the right hand side shows the overall design of Merry Porter, a robot used for waste transportation and drugs delivery in hospitals. The laser range-finder used for localization is located at the top of the front pole, whereas a system made up of a security laser scanner and a sonar ring is placed at the base level to detect people and (moving) objects outside the LFCS. It is worth noting that this decoupling is enforced by regulations and security standards for deployment of automated moving machines in human populated environments. It can be argued that the LFCS approach forces robots to be characterized by a mechanical structure able to host a sensor for localization "above human heads". This can not be guaranteed for any robot. However, a number of already available design choices seem to go towards this direction: as a matter of fact, robots used in industrial settings (such as Automated Guided Vehicles and similar machines[3]) or in hospital scenarios[4] are often characterized by analogous mechanical designs to exploit this "freely sensing perspective".

With respect to SW issues, the presence of two different sensors implies the existence of two distinct data sources: the former is used solely by the localization subsystem, whereas the second is exploited by the obstacle avoidance subsystem. As a consequence, this has significant benefits on the robot SW architecture. On the one hand, laser data do not have to be considered a shared resource any more. On the other hand, the mentioned decoupling has the effect of enforcing redundancy as well as fault-tolerance.

4.2 Feature Extraction and Matching in the LFCS

Several steps are sequentially executed, implementing a typical Split & Merge algorithm [58]. This family of line extraction methods exhibits a number of advantages if compared with other existing techniques [6, 19, 34, 41, 48, 66]: by exploiting local scan templates and acting upon smaller chunks of data, they are computationally faster and produce very accurate (although often suboptimal) results.[5]

The steps are: Preprocessing, Scan Segmentation, Line Extraction and Line Matching. Preprocessing and Scan Segmentation do not deserve further discussion. These steps include standard techniques to filter out outliers and to segment range data into "almost continuous", non overlapping, sets of points to be fed to the next computational steps. It is just worth pointing out that

[3] For instance, please check commercial solutions for the delivery of supplies by JBT Corporation at www.jbtc-agv.com.

[4] Although the robots commercialized by InTouch Health and Swisslog do not exploit the LFCS for localization, their mechanical structure appears compatible with such a solution: see RP-7*i* at www.intouchhealth.com, and RobotCourier™ at www.swisslog.com). As a counter example, consider the Swisslog's TransCar and Aethon's TUG® at www.aethon.com.

[5] Please note that the LFCS approach does not necessarily require a Split & Merge algorithm. Other choices are equally legitimate and there are no evident reasons to think that they would invalidate the following considerations.

special care is devoted to reduce a distortion effect on the raw range measurements originated by the high robot speed. Since a complete laser scan requires a non negligible time, when all range data are finally available they do not coherently represent the geometry of the environment. As a consequence, a sort of "ego motion" compensation filter is used in the Preprocessing step.

Line Extraction produces a set λ of $\mathbb{L}$ line segments, where:

$$\lambda = \{l_j\}, \quad j = 1, \ldots, \mathbb{L}. \tag{4}$$

Each segment line $l_j \in \lambda$ is characterized by a 2-element vector $(\rho_j, \alpha_j)^T$. In particular, by defining L_j the (infinite) straight line on which l_j lies, ρ_j is the distance between L_j and the robot-centered reference frame, whereas α_j is the angle between ρ_j and the robot heading. By relying on geometrical considerations, each line segment l_j is next interpreted as an *oriented* line segment. This allows for the distinction between the opposite external wall and object surfaces during the Feature Matching phase and the reduction of possible wrong associations.

For each l_j, an observation $z_k = l_j = (\rho_j, \alpha_j)^T$ is available in principle to correct the robot pose. For this purpose, according to the EKF equations, we need to compute the expected measurement $h(\hat{\chi}^-) = m_i = (\rho_i, \alpha_i)^T$ given the current a priori estimate $\hat{\chi}^-$ and the known map: the complete expression is given in Eq. 7. Since the map μ is a set of $\mathbb{M}$ oriented line segments

$$\mu = \{m_i\}, \quad i = 1, \ldots, \mathbb{M}, \tag{5}$$

computing $h(\hat{\chi}^-)$ requires to find the line segment $m_i \in \mu$ best matching l_j. Since both each $l_j \in \lambda$ and each $m_i \in \mu$ are characterized by an associated covariance matrix (see Section 5), it is possible to compute the Mahalanobis distance associated with each couple (l_j, m_i): for each l_j, the line segment m_i at minimum Mahalanobis distance is selected and the resulting couple is then used by the Kalman filter for the Measurement Update step.

4.3 State Estimation in the LFCS

In the original formulation of the Kalman filter the process f and the measurement model h are required to be linear. If this is the case, Eq. 1 can be rewritten as follows:

$$\begin{aligned} \chi_k &= A\chi_{k-1} + Bu_k + w_{k-1}, \\ z_k &= H\chi_k + v_k. \end{aligned} \tag{6}$$

Since the process model reflects the robot kinematics, in most cases it can not be modelled as linear. Furthermore, it often happens that the measurement model itself does not meet the linearity requirement. As a matter of fact, in map-based localization non-linearity shows up whenever "corners" or "line intersections" are used as features to correct the pose estimate. If either f or h (or both) are not linear, it is necessary to derive A, W, H and V, i.e., the Jacobian matrices of f and h with respect to the state χ, the process noise w and the measurement noise v. Unfortunately, the resulting extended Kalman filter (EKF) is no more optimal, and—in general—it suffers from other well known problems as a consequence of linearisation.

The EKF iterates between the Time Update step, where the estimated state $\hat{\chi}_{k-1}$ and state covariance P_{k-1} are projected to the next step through the process model (thereby producing the a priori estimates $\hat{\chi}_k^-$ and P_k^-), and the Measurement Update step, where $\hat{\chi}_k^-$ and P_k^- are updated through the available observations z_k (thereby producing the a posteriori estimates $\hat{\chi}_k$ and P_k). In the following discussion, the focus is mostly on the second step, since the first one does not depend on the adopted robot sensory system.

In order to obtain $\hat{\chi}_k$, the expected measurement $h(\hat{\chi}_k^-)$ must be compared with the actual measurement z_k. If $\hat{\chi}_k^- = (\hat{x}_k^-, \hat{y}_k^-, \hat{\theta}_k^-)^T$ is the a priori state estimate and m_i has been selected as the line segment in the map corresponding to the detected line segment l_j, the expected measurement $h(\hat{\chi}_k^-)$ is given by the following *linear* equation:

$$h(\hat{\chi}_k^-) = \begin{pmatrix} \rho_i \\ \alpha_i \end{pmatrix} = \begin{pmatrix} \dfrac{a_i\hat{x}_k^- + b_i\hat{y}_k^- + c_i}{\sqrt{a_i^2 + b_i^2}} \\ \arg(b - ia) - \hat{\theta}_k^- \end{pmatrix}. \tag{7}$$

In Eq. 7, ρ_i represents the signed distance between the robot position and the infinite straight line M_i on which m_i lies, whereas α_i (expressed in complex notation) represents the difference between the robot heading and the orientation of the signed distance ρ_i. Coefficients a_i, b_i and c_i characterize the implicit equation[6] of M_i expressed as $a_i x + b_i y + c_i = 0$, and can be computed using simple geometrical considerations. By subtracting the constant term—which is independent from the current estimate $\hat{\chi}_k^-$—both from the actual $z_k = (\rho_j, \alpha_j)^T$ and the expected measure $(\rho_i, \alpha_i)^T$, the measurement model can be written as $H\hat{\chi}_k^-$, where:

$$H = \begin{pmatrix} \frac{a_i}{\sqrt{a_i^2 + b_i^2}} & \frac{b_i}{\sqrt{a_i^2 + b_i^2}} & 0 \\ 0 & 0 & -1 \end{pmatrix}. \tag{8}$$

At this point, the EKF Measurement Update step can be performed. A number of considerations can be done.

First, the measurement model $H\hat{\chi}_k^-$ is linear. This is a consequence of the choice to characterize each observation through the vector $z = (\rho, \alpha)^T$, which—in most cases—can be reasonably modelled as corrupted by Additive White Gaussian Noise $v = N(0, R)$ [6, 34, 41, 54, 55, 58, 66]. If a different characterization of z were chosen, this could not be guaranteed. However, this choice for z is effective because, in the LFCS, the number of detectable line segments is smaller than elsewhere, each segment being “longer” and less frequently occluded by moving objects. In other words, line segments can be easily distinguished from each other.

Second, an observability analysis must be carried out. To this aim, it is required to make few assumptions on the process, which—in general—is not linear. However, it is straightforward to show that, for every possible linearisation A of f, the rank of the observability matrix

$$O = \left[\, H^T \;\; A^T H^T \;\; (A^T)^2 H^T \,\right] \tag{9}$$

is always guaranteed to be ≥ 2, since the observability space (which is a subset of $\mathbb{R}^3$) always includes the subspace defined by H rows. The rank is minimum, for example, when A results from the linearisation of a differentially driven system (a very common solution in indoor mobile robotics). When this happens, the system must rely on subsequent observations to correct all the three components of the robot state. Note in fact that, for every observed line l_j, a different measurement model H_i is considered, whose first row corresponds to the implicit equation of the infinite straight line on which m_i (at minimum Mahalanobis distance with respect to l_j) lies. If two successive observations l_j and l_m correspond to non parallel lines m_i and m_n, the first rows of H_i and H_n are linearly independent: together with the second row of either matrices they form a basis for $\mathbb{R}^3$, thus guaranteeing that all the state components can be observed.

5 Effects on the a Posteriori Covariance

Since the benefits of the LFCS approach have been discussed, possible concerns about its drawbacks could be raised.

Question 1 In the LFCS, a smaller number of features is available. Does the decreased number of observations negatively affect the localization process?

When features are extracted through an approach maximizing the a posteriori likelihood of ρ and α (such as in [55]), line segments in the LFCS are expected to be less in number but more reliable. As a matter of fact, a bigger number of range measurements contributes to each line segment. Since there is a trade off between the number of available features and the likelihood of each feature, the question is whether it is better to have a smaller set of “longer”, more reliable observations (i.e., line segments detected in the LFCS), or a bigger set of “shorter” ones (i.e., by taking range measurements at a lower height). Notice that—up to now—it has been assumed that a set λ of best fitting lines is extracted from the available range data. In principle, raw range data could be directly used to optimally estimate $\hat{\chi}$ through an EKF or

[6] The use of the implicit line equation allows for neglecting particular and degenerate cases, such as vertical infinite straight lines $x = k$, with k an arbitrary constant, which characterize other possible formulations.

another optimal estimator: as a matter of fact, adding intermediate filters for Feature Extraction and Matching can not bring any improvement in terms of accuracy. In practice, feature based methods are often preferred for different reasons, i.e., synthesis in the representation and computational efficiency.

Question 1 must be more precisely defined. If the number of features extracted from raw data is bigger (i.e., scans taken outside the LFCS), each laser scan is likely to contain more information to disambiguate between possible robot poses. This property is fundamental when performing global localization. However, global localization can not be solved through Kalman filtering, since a unimodal estimate of the robot pose is available and updated each time step.[7] Therefore, the claim that a bigger number of features helps the robot to disambiguate between different areas of the environment just makes no sense, when using an EKF. On the contrary, when dealing with position tracking, Question 1 can be rigorously answered only in terms of *state observability* and *accuracy of the state estimate*.

The observability analysis shows that two non parallel detected lines l_j and l_m are necessary and sufficient to correct all the components of the robot state $\chi = (x, y, \theta)^T$. One could argue that outside the LFCS a larger number of features is available, thus statistically increasing the likelihood of finding at least two non parallel (hopefully perpendicular) lines. However, this probability can hardly be quantified. Furthermore, when relying on features that are not permanent, it is very difficult to predict whether a given area could meet the observability requirements or not. On the contrary, features in the LFCS mostly correspond to walls and furniture that are depicted on the interior plants of buildings, thereby allowing an a priori analysis of the system behaviour on the basis of pure geometric considerations.

The accuracy of the estimate is measured by the covariance of the position error. Therefore, when focusing on accuracy, Question 1 must be re-stated as follows.

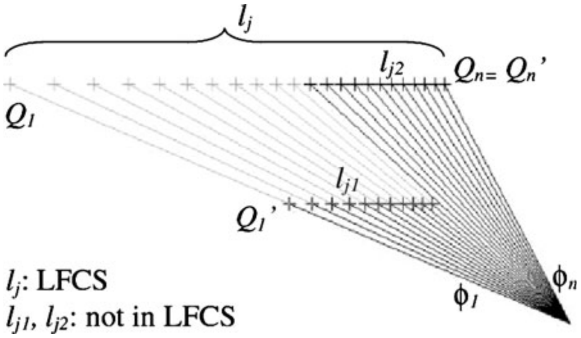


Fig. 3 A typical situation: given a scanning interval $(\phi_1, \ldots, \phi_n)$, outside the LFCS range data contribute to two line segments l_{j1} and l_{j2}, whereas in the LFCS only one line segment l_j is obtained

Question 2 A full scan is composed of a given number of range points. Let us assume that n range points contribute to a single line segment l_j. Would the a posteriori error covariance (computed using the EKF) be lower, if the n range measurements produced two line segments l_{j1}, l_{j2} parallel to l_j, instead of one single line segment?

When considering localization accuracy, problems related to observability can be ignored by comparing parallel line features (Fig. 3). In this situation, line segments can be "scored" on the basis of their "correcting power". Since, according to the EKF equations, the Kalman gain K in the Measurement Update step inversely depends on the covariance R of the updating observation $z = (\rho, \alpha)^T$, the effect of one single observation with smaller covariance R_j (corresponding to l_j) can be compared with the joint effect of a couple of observations with higher covariances R_{j1} and R_{j2} (corresponding, respectively, to l_{j1} and l_{j2}).[8] The procedure can be iterated to consider an arbitrary number of observations.

In principle, when using two observations in cascade to update the filter (l_{j1} and l_{j2} extracted from the same scan), one should consider the evolution of the state during the acquisition, extraction and matching phase for each feature. As a matter of fact, this is not negligible especially when the robot moves at high speed. In practice, after opportunely processing range measure-

[7] EKF owes its success to the fact that self-localization and SLAM can be often treated as position tracking problems.

[8] When R is smaller, then K is bigger and the decrement of the a posteriori covariance P of $\hat{x}$ is higher.

ments, it is common to consider the robot as if it were still while line segments extracted from the same scan are fed to the filter:

1. Between subsequent scans, the estimated state $\hat{\chi}$ and the covariance P are updated in just one step through the process model A that reflects the (possibly linearised) robot kinematics;
2. When a new scan is available, $\hat{\chi}$ and P are updated by considering all the available observations and assuming no robot motion (i.e., $\hat{\chi}_k^- = \hat{\chi}_{k-1}$ and $P_k^- = P_{k-1}$).

To allow for a comparison between the "correcting power" of l_j and the couple of observations l_{j1} and l_{j2}, the following Theorem is introduced.

Theorem 1 *Given that P_k^- is the estimated state covariance immediately after the Time Update step of the EKF at the kth iteration. Given that $P_{k+1}^- = P_k$, i.e., the estimated covariance does not change during the Time Update step at the $k+1$th iteration. Given two subsequent observations z_k and z_{k+1} with covariances R and Q, used to update the estimate at the kth and $k+1$th iterations. Then the two observations produce the same a posteriori estimated state covariance P_{k+1} than a single observation with covariance Z such that*

$$Z = R\left(R+Q\right)^{-1}Q. \tag{10}$$

Proof See Appendix. □

When covariances R_j, R_{j1}, and R_{j2} are known (e.g., we are dealing with three specific features l_j, l_{j1} and l_{j2}), Theorem 1 can be used to compare their effect on the state covariance P, e.g., by computing the eigenvalues of R_j and $R_{j1,j2} = R_{j1}\left(R_{j1}+R_{j2}\right)^{-1}R_{j2}$, and by comparing the corresponding ellipsoids. However, if one wants a general rule to decide whether it is better to aggregate range points into "longer" or "shorter" features, this is not sufficient. To this aim, it is necessary to investigate in depth how the covariance of each feature depends on the covariances of the contributing range measurements. Therefore, *Question 2* requires to answer the following question.

Question 3 Given a set of n range measurements expressed in polar coordinates $(d,\phi)^T$, where d and ϕ are the current range and scan angle, and σ_d and σ_ϕ are the corresponding variances, would the "correcting power" be higher if the set of range measurements were split into two subsets contributing respectively to two parallel line segments l_{j1} and l_{j2} subtending the same scanning interval $(\phi_1,\ldots,\phi_n)$ as l_j?

In order to answer *Question 3*, it is necessary to accurately model the observation covariance R_j as a function of the covariances of all the range points contributing to l_j. To this aim, we refer to [55], where a maximum likelihood approach that provides formulas for computing the line segment covariance starting from individual weightings of range points is introduced. Each scan point q_l is first defined with respect to the robot-centered reference frame, thereby taking the form:

$$q_l = d_l\begin{pmatrix}\cos\phi_l\\ \sin\phi_l\end{pmatrix}, \tag{11}$$

whereas the corresponding covariance Q_l is approximated as follows:

$$Q_l = \frac{d_l^2\sigma_\phi^2}{2}\begin{pmatrix}2\sin^2\phi_l & -\sin 2\phi_l\\ -\sin 2\phi_l & 2\cos^2\phi_l\end{pmatrix} + \frac{\sigma_d^2}{2}\begin{pmatrix}2\cos^2\phi_l & \sin 2\phi_l\\ \sin 2\phi_l & 2\sin^2\phi_l\end{pmatrix}. \tag{12}$$

Afterwards, a coordinate frame RS_j (Fig. 4) associated with each observation $z_j = \left(\rho_j, \alpha_j\right)^T$ is introduced. RS_j corresponds to the robot-centered reference frame rotated by α_j (i.e., with R perpendicular and S parallel to l_j). The covariance

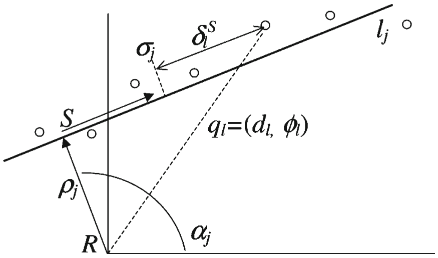


Fig. 4 The covariance R_j of a line feature l_j depends on the covariances Q_l of all contributing scan points q_l

$Q_l^{RS_j}$ associated with the lth range point in the new frame RS_j is defined as:

$$Q_l^{RS_j} = \begin{pmatrix} Q_l^R & 0 \\ 0 & Q_l^S \end{pmatrix}, \tag{13}$$

where $Q_l^{RS_j}$ can be expressed as a function of Q_l as follows:

$$\begin{aligned} Q_l^R &= Q_{l,11}\cos^2\alpha_j + 2Q_{l,12}\sin\alpha_j\cos\alpha_j \\ &\quad + Q_{l,22}\sin^2\alpha_j, \\ Q_l^S &= Q_{l,11}\sin^2\alpha_j - 2Q_{l,12}\sin\alpha_j\cos\alpha_j \\ &\quad + Q_{l,22}\cos^2\alpha_j. \end{aligned} \tag{14}$$

When computing ρ_j and α_j, the observation covariance R_j finally results as follows:

$$R_j = \begin{pmatrix} R_{\rho\rho} & R_{\rho\alpha} \\ R_{\alpha\rho} & R_{\alpha\alpha} \end{pmatrix}, \tag{15}$$

where:

$$\begin{aligned} R_{\rho\rho} &= \left(\sum_{l=1}^{n} \frac{1}{Q_l^R}\right)^{-1}, \\ R_{\alpha\alpha} &= \left(\sum_{l=1}^{n} \frac{\delta_l^{S\,2}}{Q_l^R}\right)^{-1}, \\ R_{\rho\alpha}, R_{\alpha\rho} &= -R_{\rho\rho}R_{\alpha\alpha}\sum_{l=1}^{n}\frac{\delta_l^S}{Q_l^R}, \end{aligned} \tag{16}$$

and

$$\delta_l^S = q_l^S - \sigma. \tag{17}$$

In Eq. 17, δ_l^S measures the distance (computed along S) of the lth range point from the weighted mean σ of all the points contributing to l_j:

$$\sigma = \left(\sum_{l=1}^{n}\frac{1}{Q_l^S}\right)^{-1}\sum_{l=1}^{n}\frac{q_l^S}{Q_l^S}. \tag{18}$$

Let us temporarily assume that the observation vector z contains only the observed distance ρ of each line segment, and consequently let us ignore the line segment orientation α. In this case, the covariance matrix R_j in Eq. 15 coincides with element $R_{\rho\rho}$. As discussed, R_j comes from n contributing range points q_l, each one with covariance $Q_l^{RS_j}$: if the set of n range points is split into two subsets, containing respectively n_1 and $n - n_1$ range points,[9] the two subsets can be used to extract two lines segments l_{j1} and l_{j2} with covariances R_{j1} and R_{j2}, respectively. From Eq. 16:

$$\begin{aligned} R_{\rho\rho 1} &= \left(\sum_{l=1}^{n_1}\frac{1}{Q_l^R}\right)^{-1}, \\ R_{\rho\rho 2} &= \left(\sum_{l=n_1+1}^{n}\frac{1}{Q_l^R}\right)^{-1}, \end{aligned} \tag{19}$$

and from Eq. 10:

$$\begin{aligned} R_{\rho\rho 1,2} &= \frac{1}{\sum_{l=1}^{n_1}\frac{1}{Q_l^R}} \frac{\sum_{l=1}^{n_1}\frac{1}{Q_l^R}\sum_{l=n_1+1}^{n}\frac{1}{Q_l^R}}{\sum_{l=n_1+1}^{n}\frac{1}{Q_l^R} + \sum_{l=1}^{n_1}\frac{1}{Q_l^R}} \frac{1}{\sum_{l=n_1+1}^{n}\frac{1}{Q_l^R}} \\ &= \left(\sum_{l=1}^{n_1}\frac{1}{Q_l^R} + \sum_{l=n_1+1}^{n}\frac{1}{Q_l^R}\right)^{-1} = R_{\rho\rho}. \end{aligned} \tag{20}$$

As expected, if each range point contributing to l_j had the same covariance Q_l of a corresponding point in l_{j1} or l_{j2}, the a posteriori estimated state covariance would result identical after observing l_j or the couple l_{j1} and l_{j2} (remember that Q_l^R in Eq. 20 is a function of solely Q_l and α_j). According to Eq. 12, the covariance Q_l varies with the range d_l associated with each scan angle ϕ_l. As a consequence, Eq. 20 does not help to compare situations like Fig. 3. This problem will be soon considered. For the moment, note that the result is applicable to situations like in Fig. 5: the covariance of the $x-$ and $y-$ components of $\hat{\chi}$ (the ones observed through ρ) does not increase or decrease when an occluding object (i.e., outside the LFCS) interrupts the sequence of collinear range points (by ignoring the contribution of the range measurements that return the distance to the occluding object itself). By reconsidering the more general case of Fig. 3, it is possible to note that, if the angular error σ_ϕ on range measurements is small enough to be ignored, the covariance Q_l does not depend on d_l for Eq. 12 and $Q'_l \approx Q_l$ holds. Under these conditions, which are often

[9] n_1 is a generic number between 1 and n. The properties discussed in the following paragraphs are independent from the actual value of n_1.

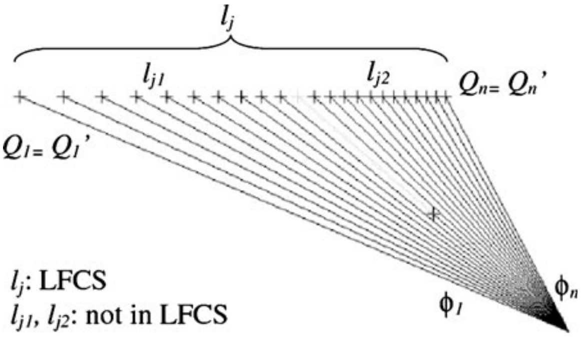


Fig. 5 Another typical situation: a small occluding object causes two line segments l_{j1} and l_{j2} to be observed outside the LFCS, whereas only one line segment l_j is detected in the LFCS

verified, Eq. 20 holds for any triplet of parallel line segments l_j, l_{j1}, l_{j2} which subtend the same scanning interval $(\phi_1, \ldots, \phi_n)$.

If the observation vector z contains also the line orientation α, but errors on ρ and α are assumed to be uncorrelated (which, in general, is not true), then $R_{\rho\alpha} = R_{\rho\alpha 1} = R_{\rho\alpha 2} = R_{\rho\alpha 1,2} = 0$ and Eq. 10 can be applied to $R_{\rho\rho}$ and $R_{\alpha\alpha}$ independently. For $R_{\alpha\alpha 1,2}$ a result similar to Eq. 20 is obtained:

$$R_{\alpha\alpha 1,2} = \left(\sum_{l=1}^{n_1} \frac{\delta_l^{S^2}}{Q_l^R} + \sum_{l=n_1+1}^{n} \frac{\delta_l^{S^2}}{Q_l^R} \right)^{-1}. \tag{21}$$

In general, $R_{\alpha\alpha} \neq R_{\alpha\alpha 1,2}$, since the values of δ_l^S in Eq. 21 are different from the corresponding values in Eq. 16. This is a direct consequence of Eq. 17: the weighted mean σ must now be computed separately for scan points in l_{j1} and l_{j2}, thus yielding $\sigma_{j1} \neq \sigma_{j2}$. If, once again, one considers parallel line segments subtending the same scanning interval $(\phi_1, \ldots, \phi_n)$, $R_{\alpha\alpha 1,2}$ is necessarily bigger than $R_{\alpha\alpha}$ (and hence the "correcting power" is lower), since the distance δ_l^S of each range point in l_{j1} and l_{j2} from the corresponding σ is smaller. This is true both for the situation in Fig. 3 and, if the angular error σ_ϕ on each range measurement can be ignored, for the situation in Fig. 5.

From these considerations, one could be tempted to conclude that the a posteriori state covariance of the θ component of $\hat{\chi}$ is expected to be lower, when range measurements are such that longer and farther line segments can be extracted. Unfortunately, these conclusions are not valid whenever ρ and α are correlated. In order to provide a more general answer taking the effects of correlation into account, a statistical analysis has been performed in simulation, which is presented in Section 6.2. However, it is possible already to provide an answer to Question 3, and to backpropagate such answer to Questions 2 and 1.

Answer to Question 3 In general, a single line segment in the LFCS has a "correcting power" that is not lower than two lines outside the LFCS (actually, when ρ and α are "almost uncorrelated", its "correcting power" is higher).

Answer to Question 2 In general, a single observation in the LFCS does not produce an error covariance that is higher than two observations outside the LFCS.

Answer to Question 1 In general, considering also the observability analysis, the reduced number of localization features in the LFCS do not affect negatively the localization process.

Finally note that, since all the involved quantities are computed according to the maximum likelihood approach described in [55], these conclusions are obviously valid for every line fitting algorithm estimating $z = (\rho, \alpha)^T$ optimally.

6 Experimental Results

6.1 Minimization of the Unevenness Index

Experiments have been performed to search for a Low Frequency Cross-Section of the environment where laser scans are more "regular" and "static". In particular, even though minimization of U has been performed by considering all the possible configurations in the height range $0 \div 220$ cm with a 10 cm resolution, pictures show results only for three different representative height values (see Fig. 1): 10 cm, 90 cm, and roughly 2 m. We decided to synthesize—for ease of representation—the whole range of possible heights into three classes or *views*: *bottom view*, *mid view*, and *top view*. About one thousand scans have been collected in different places of a typical office-like environment (our department), a house and a garage. Each scan lasts about five minutes, thereby requiring to average results over time. Results do not significantly vary when the laser sensor is moved up or down 20 or 30 cm, thereby allowing us to consider only three *views*.

In all the room-like environments (Fig. 6 on the top), the unevenness index U, averaged over five minutes, exhibits a similar behaviour in bottom and mid views (first and second column), whereas in the top view (third column) it is significantly smaller. Moreover, the variance over time is significant in the bottom and mid views (fourth and fifth columns), whereas it is almost zero in the top view (sixth column; variance is not null because of sensor noise). The pictures on the right column of the Figure depict the profile of U over time. In corridor-like environments (Fig. 6 in the middle), experiments show that the three views are comparable, due to the fact that corridors are transition places where objects do not "tend to accumulate". Nonetheless, the variance over time of U is significant only in the bottom and mid views, since scans are affected by the presence of people. Another class of experiments with a longer duration (about three hours, see Fig. 6 in the bottom) has been performed. However, even when U is higher when computed in the top view, it is nevertheless more robust with respect to environmental changes.

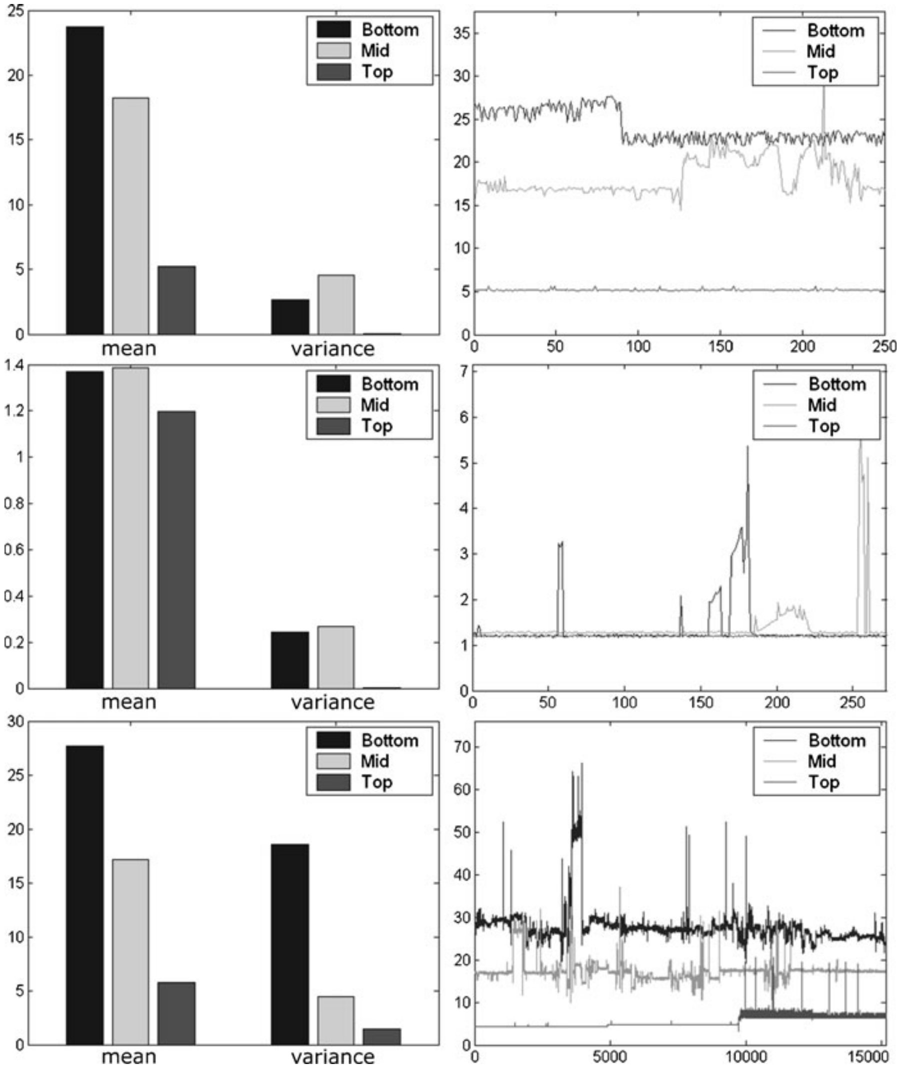


Fig. 6 *Left column*: three examples of U values (both mean values and variances of *top*, *mid* and *bottom* views are reported). *Right column*: the corresponding temporal evolutions of U

Table 1 Mean values and standard deviations computed for Υ

	Rooms		Corridors	
	Mean	Std	Mean	Std
$\Upsilon_{\text{top,bottom}}$	0.31	0.17	1.03	0.3
$\Upsilon_{\text{top,mid}}$	0.65	0.2	1.05	0.24
$\Upsilon_{\text{mid,bottom}}$	0.50	0.22	0.99	0.22

Table 1 shows the mean value and the standard deviation of the unevenness ratio Υ introduced in Eq. 3, averaged over about one thousand laser scans in different environments. The results confirm what was already evident in Fig. 6: in rooms always happens that $\Upsilon_{\text{top,bottom}} < 1$ and $\Upsilon_{\text{top,mid}} < 1$, whereas in corridors $\Upsilon_{\text{top,bottom}} \simeq \Upsilon_{\text{top,mid}} \simeq 1$. It is worth noting that even if in corridor-like environments the position of the laser range-finder is almost indifferent, in most situations it is not: the existence of a Low Frequency Cross-Section is evident, thus validating the whole approach.

The unevenness index U is compared with other indicators: Average Roughness R_a, Root-Mean-Square Roughness R_q, Total Roughness R_t, Maximum Roughness Height R_y and High Spot Count HSC. Tests performed show that all indicators are very often in agreement, i.e., they capture the same qualitative idea of "regularity of a scan profile".

6.2 Correcting Power of Line Features

Experiments in simulation using MATLAB have been performed: in each experiment, a line l_j taken in the LFCS is compared with a couple of lines l_{j1}, l_{j2} parallel to l_j subtending the same scanning interval $(\phi_1, \ldots, \phi_n)$. Specifically, given the covariance matrices R_{j1} and R_{j2}, the covariance $R_{j1,j2}$ corresponding to applying two corrections in cascade is computed according to Eq. 10. Next, the eigenvalues of R_j and $R_{j1,j2}$ are considered, and used to compute the ratio $R_{j/j1,j2}$ between the areas of the two corresponding ellipsoids. The results are interpreted as follows: when $R_{j/j1,j2} < 1$, the "correcting power" of l_j is higher, otherwise it would have been better to observe l_{j1} and l_{j2} in cascade.

In these simulations, a laser range finder scanning 180 deg with a resolution of 0.5 deg is considered. According to [34], errors in each simulated range-finder measurement are modelled with standard deviations $\sigma_d = 26$ mm and $\sigma_\phi = 0.125$ deg. Moreover, $R_{j/j1,j2}$ is computed for many different triplets l_j, l_{j1}, l_{j2}, by varying most parameters involved: (1) the distance ρ_j between l_j and the robot; (2) the lateral displacement σ_j of l_j according to Eq. 18; (3) the length ϵ_j of l_j; (4) the distance ρ_{j1} between l_{j1} and the robot, with the constraint $\rho_{j1} < \rho_j$; (5) the ratios n_1/n and n_2/n, i.e., the subset of measurements contributing—respectively—to l_{j1} and l_{j2}. In all the experiments, $\rho_{j2} = \rho_j$: that is, it is assumed that l_{j1} corresponds to an occluding object (thus being closer to the robot, as in Fig. 3), whereas l_{j2} corresponds to a portion of wall or furniture that is visible at whatsoever height from the floor. Finally, all simulations assume $\alpha_j = 0$, since the covariance of observed features is invariant with respect to robot rotations, and therefore the robot orientation can be ignored.

The analysis in simulation confirms theoretic results. Whether it is preferable a smaller set of "longer" line segments *or* a bigger set of "shorter" ones, depends on different factors, involving both the geometric characteristics of the environment and the robot pose. However, summarized results for $R_{j/j1,j2}$ shown in Tables 2 and 3 allow to claim that, on average, the LFCS is expected to improve localization accuracy in a standard indoor environment with 1 m $< \rho_j, \rho_{j1}, \rho_{j2} <$ 10 m and 2 m $< \epsilon_j <$ 10 m. Tables 2 and 3 must be read as

Table 2 Summarized results for $R_{j/j1,j2}$ (I)

	2–1	4–1	6–1	8–1	10–1	12–1	4–2	6–2	8–2	10–2	12–2
2 m	0.31	0.24	0.24	0.25	0.25	0.25	0.26	0.25	0.25	0.25	0.25
4 m	0.43	0.26	0.23	0.24	0.25	0.25	0.28	0.24	0.25	0.25	0.25
6 m	0.64	0.33	0.25	0.24	0.25	0.25	0.35	0.26	0.25	0.25	0.26
8 m	0.91	0.44	0.30	0.26	0.25	0.25	0.48	0.32	0.27	0.26	0.26
10 m	1.25	0.60	0.38	0.3	0.27	0.26	0.65	0.40	0.31	0.28	0.27

Table 3 Summarized results for $R_{j/j1,j2}$ (II)

	6–4	8–4	10–4	12–4	8–6	10–6	12–6	10–8	12–8	12–10
2 m	0.29	0.28	0.27	0.26	0.31	0.29	0.28	0.32	0.30	0.32
4 m	0.27	0.27	0.27	0.27	0.29	0.29	0.28	0.31	0.30	0.33
6 m	0.29	0.27	0.27	0.27	0.29	0.29	0.28	0.31	0.30	0.32
8 m	0.35	0.29	0.27	0.27	0.31	0.29	0.29	0.31	0.30	0.32
10 m	0.44	0.33	0.29	0.28	0.35	0.31	0.29	0.33	0.31	0.33

follows: each row corresponds to a different value for ϵ_j ranging from 2 m to 10 m; each column corresponds to a different distance range $\rho_{\max} - \rho_{\min}$, with $\rho_{j1} \leq \rho_{j2} = \rho_j = \rho_{\max}$; cells contain the mean value of $R_{j/j1,j2}$ mediated over ρ_{j1} varying between $\rho_{\min}$ and $\rho_{\max}$, σ_j varying between −5 and 5 m, n_1/n varying between 0.2 and 0.8.

6.3 Experiments with Simulated and Real Robots

To validate the LFCS approach, experiments have been performed both in simulation and with real robots. In particular, each experiment is repeated twice, by providing the robot with two different maps, one representing the world as seen in the LFCS, and the other corresponding to a lower height ("not LFCS" in the following paragraphs).

Experiments in simulation are performed in a room such as the one in Fig. 7. Different sizes are considered, ranging from 10 × 10 m^2 to 80 × 80 m^2. The external, light-gray perimeter of the room corresponds to walls detected in the LFCS, whereas the inner contour corresponds to occluding objects detected when the laser is set at a lower height. Experiments in simulation are particularly significant since the very same world model given to the robot is used to simulate laser range measurements (sensor noise is added). On the contrary, with real robots, the map is built by hand, and therefore its accuracy depends also on one ability to create a consistent map. In addition, in a very messy and cluttered environment (such as the lab environment in Fig. 1), it is very difficult to find straight line segments at all, thereby making localization even harder. Both in simulation experiments and with real robots, the LFCS world model is obviously simpler, with a smaller number of "longer" features. In simulation, on average, two or three features per scan are usually extracted and matched in the LFCS, whereas 18 or 19 features are observed outside the LFCS. With real robots, 5 or 6 features per scan are observed in the LFCS, 14 or 15 elsewhere.

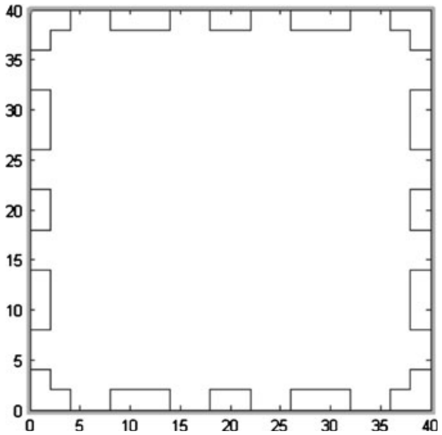


Fig. 7 A 40 × 40 m room used in experiments in simulation: the LFCS map corresponds to the external perimeter, whereas the map outside the LFCS comprises also small objects close to the walls

Figure 8 shows the error convergence (x, y and θ components) in simulation with a still robot located in a 40 × 40 m^2 room (scans are taken and processed at a 4 Hz frequency). In all the experiments with a still robot, convergence in the LFCS is significantly faster. Moreover, the effects of wrong feature association, i.e., an increment in positioning errors, can be sometimes observed in the "not LFCS" plots (e.g., θ component in the second column). Incorrect matching, when iterated, can lead to dramatic localization failures, from which it is hard to recover. Within the LFCS, wrong associations have not been reported, due to the intrinsically simple structure of the environment.

Simulated experiments with a moving robot have been performed as well, by purposely introducing errors in the odometric reconstruction. The simulated robot moves in the same set of

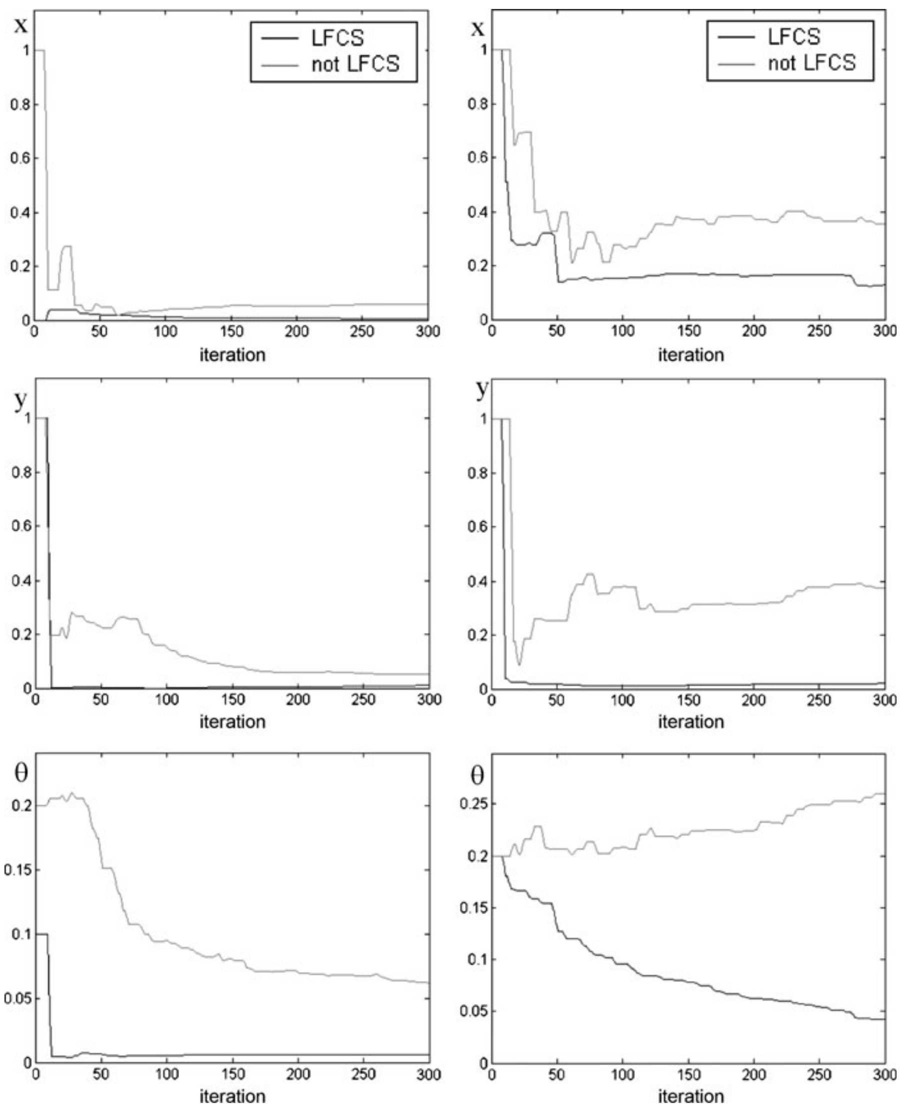


Fig. 8 Examples of positioning error in meters (i.e., distance between real and estimated position) over time with a still robot in simulation: the error is always lower when localization is performed in the LFCS

rooms already considered in the previous experiments, and the positioning error during motion is recorded and averaged for the two cases “LFCS” and “not LFCS”. Table 4 summarizes results for x, y, and θ components of the error, showing that—on average—a bigger accuracy is obtained in the LFCS.

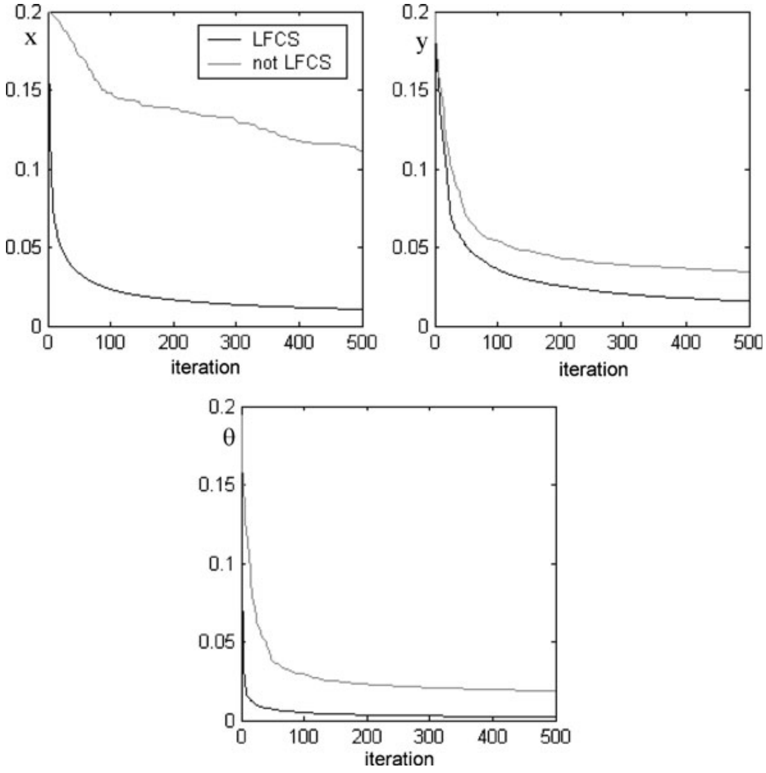
Table 4 Positioning error during motion [m]

		10 × 10	20 × 20	40 × 40	80 × 80
x	Not LFCS	0.070	0.115	0.164	0.094
	LFCS	0.060	0.052	0.061	0.078
y	Not LFCS	0.074	0.117	0.067	0.080
	LFCS	0.062	0.058	0.080	0.056
θ	Not LFCS	0.036	0.047	0.036	0.090
	LFCS	0.015	0.012	0.009	0.080

Experiments with real robots have been performed as well. The experiment shown in Fig. 9 replicates the experiment in Fig. 8 by showing the positioning error within an office-like environment at our department. Note that, in this case, convergence in the LFCS is extremely faster also for the difficulty—at a lower height—to find features stable enough and easily identifiable to be mapped.

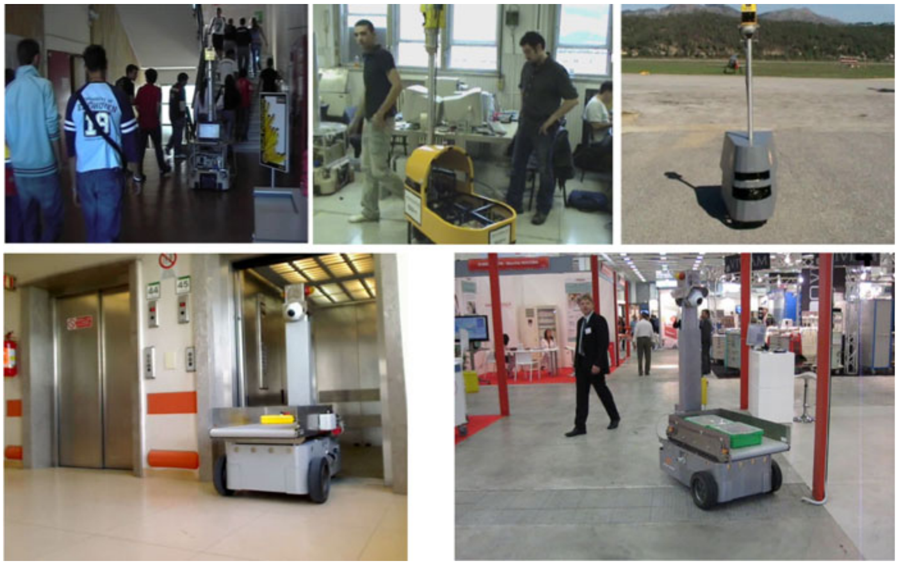
Other experiments have been performed to validate the performance of the LFCS approach.

Fig. 9 Examples of positioning error in meters (i.e., distance between real and estimated position) over time with a real still robot: the error is always lower when localization is performed in the LFCS



During the last years, LFCS localization has been applied to four different real robots (Fig. 10): a modified TRC Labmate platform and the three robots Staffetta, ANSER and MerryPorter (designed respectively for indoor service applications, outdoor surveillance, and indoor/outdoor trans-

Fig. 10 *Top*: TRC Labmate, Staffetta and ANSER. *Bottom*: Merry Porter



portation) [59]. All the robots are equipped with a SICK LMS200 placed on top of a pole about 2 m high and operate in two different areas: TRC Labmate and Staffetta navigate within the ground and second floor of our department, ANSER operates at Villanova d'Albenga Airport, whereas MerryPorter operates at the Hospital Policlinico of Modena.

Results in indoor have been collected in tens of experiments. The robots are requested to navigate autonomously at a constant speed of 0.6 m/sec. In all the experiments, robots are able to localize for almost 2 h (currently, the maximum time allowed by batteries), even if they are almost continuously surrounded by students (the entrance hall of the department building, Fig. 10 on the left) or if they operate in an extremely messy area (such as the Robotics Lab, Fig. 10 in the middle). Specifically, during all the real world experiments, an average positioning error of less than 2.5 cm is recorded. Notice that, in this case, repeatability in reaching targets is measured instead of the absolute positioning error, which explains the fact that the mean value is lower than in simulation, where the error is computed along the whole robot trajectory. An indirect indicator of the maximum error during navigation is provided also by the ability to traverse narrow doorways even in absence of obstacle avoidance algorithms: errors higher than 5 cm would cause robots to collide, which never happens even when they are moving at 0.6 m/sec. To compare performance, experiments with the laser at a lower height have been performed as well. In this case, when the environment significantly changes during motion, the localization system—as expected—is no more reliable, and it soon leads the vehicle to a dangerous, often unrecoverable configuration.

7 Conclusions

The article shows that map-based localization through a laser range-finder in cluttered and crowded environments can be improved by exploiting the existence of a Low Frequency Cross-Section, i.e., a 2D plane *parallel to the floor* where the environment slowly changes, both temporally and spatially. To quantify these concepts, an unevenness index U is introduced, which is able to attribute a score to the "roughness" and the "variability" of laser scans: low U values indicate regular and static scan profiles, which is a very desirable property for Feature Extraction and Matching (either if the map is given a priori or continuously updated through observations as in SLAM).

Specifically, it has been shown that interesting properties hold when correcting the robot pose with observations made in the LFCS. Moving objects do not to interfere with sensor data acquisition, therefore limiting occlusions and increasing the probability of reliable Feature Association in the localization process. Furthermore, the resulting map comprises a smaller number of features, therefore reducing both memory requirements and the computational complexity of localization algorithms. One is encouraged to choose straight infinite lines as features to be extracted from raw data, thereby allowing to model observations as a linear function of the state affected by Additive White Gaussian Noise (i.e., the system is very close to meeting the ideal requirements of the linear KF). The a posteriori covariance of the estimate converges faster (on average), as a consequence of the fact that range measurements can be aggregated to produce a small number of very reliable features.

Long duration localization experiments, performed both in simulation and with robots in different areas, demonstrate that the approach is very robust and precise, even when moving at high speed (0.6 m/sec) in highly crowded environments.

Appendix

In the following paragraphs Eq. 10 is formally derived. It is shown that, with respect to an EKF based localization framework, when $P^{-}_{k+1} = P_k$ (referred to as P_{k+1} and P_k afterwards), two observations with covariances R and Q produce the same a posteriori state covariance than a single observation with covariance $R\left(R + Q\right)^{-1} Q$.

If H is the linearised measurement model (the dependency upon the discrete time instant k is omitted), we can introduce M for ease of representation as follows:

$$M = \left(H P_k H^T + R\right). \tag{22}$$

Therefore, the expression of the Kalman gain K_{k+1} can be written as:

$$K_{k+1} = P_k H^T \left(H P_k H^T + R\right)^{-1} = P_k H^T M^{-1}. \tag{23}$$

The a posteriori state covariance P_{k+1} can be written as:

$$\begin{aligned} P_{k+1} &= \left(I - K_{k+1} H\right) P_k \\ &= \left(I - P_k H^T \left(H P_k H^T + R\right)^{-1} H\right) P_k \\ &= \left(I - P_k H^T M^{-1} H\right) P_k. \end{aligned} \tag{24}$$

The Kalman gain K_{k+2} at the second step is:

$$K_{k+2} = \underbrace{P_{k+1} H^T}_{K_{k+2,1}} \underbrace{\left(H P_{k+1} H^T + Q\right)^{-1}}_{K_{k+2,2}}, \tag{25}$$

where

$$\begin{aligned} K_{k+2,1} &= \left(I - P_k H^T M^{-1} H\right) P_k H^T \\ &= P_k H^T - P_k H^T M^{-1} H P_k H^T \\ &= P_k H^T \left(I - M^{-1} H P_k H^T\right) \\ &= P_k H^T \left(I - M^{-1} H P_k H^T \right. \\ &\qquad \left. - M^{-1} R + M^{-1} R\right) \\ &= P_k H^T \left(I - M^{-1} M + M^{-1} R\right) \\ &= P_k H^T M^{-1} R, \end{aligned} \tag{26}$$

and

$$\begin{aligned} K_{k+2,2} &= \left(H(I - P_k H^T M^{-1} H) P_k H^T + Q\right)^{-1} \\ &= (H P_k H^T - H P_k H^T M^{-1} H P_k H^T + Q)^{-1} \\ &= \left(H P_k H^T - H P_k H^T M^{-1} H P_k H^T \right. \\ &\qquad + H P_k H^T M^{-1} R \\ &\qquad \left. + H P_k H^T M^{-1} R + Q\right)^{-1}. \end{aligned} \tag{27}$$

By collecting $H P_k H^T M^{-1}$ we obtain from Eq. 27:

$$\begin{aligned} K_{k+2,2} &= \left(H P_k H^T - H P_k H^T M^{-1} M \right. \\ &\qquad \left. + H P_k H^T M^{-1} R + Q\right)^{-1} \\ &= \left(H P_k H^T M^{-1} R + Q\right)^{-1} \\ &= \left(\left(H P_k H^T M^{-1} + Q R^{-1}\right) R\right)^{-1} \\ &= R^{-1} \left(H P_k H^T M^{-1} + Q R^{-1}\right)^{-1} \\ &= R^{-1} \left(H P_k H^T M^{-1} + Q R^{-1} M M^{-1}\right)^{-1} \\ &= R^{-1} \left(\left(H P_k H^T + Q R^{-1} H P_k H^T \right.\right. \\ &\qquad \left.\left. + Q R^{-1} R\right) M^{-1}\right)^{-1} \\ &= R^{-1} M \left(\left(I + Q R^{-1}\right) H P_k H^T + Q\right)^{-1}. \end{aligned} \tag{28}$$

If we apply Eqs. 26 and 27 to Eq. 25 we obtain:

$$\begin{aligned} K_{k+2} &= P_k H^T M^{-1} R R^{-1} M \\ &\quad \times \left((I + Q R^{-1}) H P_k H^T + Q\right)^{-1} \\ &= P_k H^T \left((I + Q R^{-1}) H P_k H^T + Q\right)^{-1}. \end{aligned} \tag{29}$$

For ease of representation, N is introduced as follows:

$$N = (I + Q R^{-1}). \tag{30}$$

Then, the a posteriori state covariance P_{k+2} can be computed from Eqs. 24 and 29 as:

$$\begin{aligned}
P_{k+2} &= (I - K_{k+2}H)P_{k+1} \\
&= \left(I - P_kH^T(NHP_kH^T + Q)^{-1}H\right) \\
&\quad \times (I - P_kH^TM^{-1}H)P_k \\
&= \Big(I - P_kH^T(NHP_kH^T + Q)^{-1}H \\
&\quad - P_kH^TM^{-1}H + P_kH^T \\
&\quad \times \left(NHP_kH^T + Q\right)^{-1} HP_kH^TM^{-1}H\Big)\, P_k \\
&= \left(I - P_kH^T(NHP_kH^T + Q)^{-1}\right. \\
&\quad \times \left(I + (NHP_kH^T + Q)M^{-1}\right. \\
&\quad \left.\left. - HP_kH^TM^{-1}\right) H\right) P_k \\
&= \left(I - P_kH^T(NHP_kH^T + Q)^{-1}\right. \\
&\quad \times \left(I + NHP_kH^TM^{-1}\right. \\
&\quad \left.\left. + QM^{-1} - HP_kH^TM^{-1}\right) H\right) P_k \\
&= \left(I - P_kH^T(NHP_kH^T + Q)^{-1}\right. \\
&\quad \times \left(I + QR^{-1}HP_kH^TM^{-1}\right. \\
&\quad \left.\left. + QR^{-1}RM^{-1}\right) H\right) P_k \\
&= \left(I - P_kH^T(NHP_kH^T + Q)^{-1} \times\right. \\
&\quad \left. \times (I + QR^{-1}MM^{-1})H\right)P_k \\
&= \Big(I - \underbrace{P_kH^T(NHP_kH^T + Q)^{-1}}_{P_{k+2,1}} NH\Big)P_k,
\end{aligned} \tag{31}$$

where

$$\begin{aligned}
P_{k+2,1} &= P_kH^T\left(NN^{-1}NHP_kH^T + NN^{-1}Q\right)^{-1} \\
&= P_kH^T\left(N\left(HP_kH^T + N^{-1}Q\right)\right)^{-1} \\
&= P_kH^T\left(HP_kH^T + N^{-1}Q\right)^{-1}N^{-1}.
\end{aligned} \tag{32}$$

Since, from Eq. 30:

$$N^{-1} = (RR^{-1} + QR^{-1})^{-1} = R(R + Q)^{-1}, \tag{33}$$

we obtain:

$$\begin{aligned}
P_{k+2} &= \left(I - P_kH^T(HP_kH^T + N^{-1}Q)^{-1}H\right)P_k \\
&= \left(I - P_kH^T\left(HP_kH^T + R(R+Q)^{-1}Q\right)^{-1}H\right) \\
&\quad \times P_k.
\end{aligned} \tag{34}$$

Notice that the expression of P_{k+2} in Eq. 34 corresponds to the state covariance computed after one single step in Eq. 24, by opportunely substituting the covariance R with $R(R+Q)^{-1}Q$.

References

1. Adams, M., Probert, P.: The interpretation of phase and intensity data from AMCW light detection sensor for reliable ranging. Int. J. Rob. Res. **15**(5), 441–458 (1996)
2. Antonelli, G., Chiaverini, S., Fusco, G.: A calibration method for odometry of mobile robots based on the least-squares technique: theory and experimental validation. IEEE Trans. Robot. **21**(5), 994–1004 (2005)
3. Arleo, A., Millan, J., Floreano, D.: Efficient learning of variable-resolution cognitive maps for autonomous indoor navigation. IEEE Trans. Robot. Autom. **15**(6), 990–1000 (1999)
4. Armesto, L., Ippoliti, G., Longhi, S., Tornero, J.: FastSLAM 2.0: least-squares approach. In: Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006). Beijing, China (2006)
5. Baltzakis, H., Trahanias, P.: A hybrid framework for mobile robot localization: formulation using switching state-space models. Auton. Robots **15**, 169–191 (2003)
6. Bennewitz, M., Burgard, W., Thrun, S.: Using EM to learn motion behaviors of persons with mobile robots. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2002). Lausanne, Switzerland (2002)
7. Bennewitz, M., Stachniss, C., Behnke, S., Burgard, W.: Utilizing reflection properties of surfaces to improve mobile robot localization. In: Proceedings of the 2009 IEEE International Conference on Robotics and Automation (ICRA 2009). Kobe, Japan (2009)
8. Blanco, J., Gonzalez, J., Fernandez-Madrigal, J.: Consistent observation grouping for generating metric-topological maps that improves robot localization. In: Proceedings of the 2006 International Conference on Robotics and Automation (ICRA 2006). Orlando, FL (2006)
9. Borenstein, J.: Experimental results from internal odometry error correction with the OmniMate mobile robot. IEEE Trans. Robot. Autom. **14**(6), 963–969 (1998)
10. Bosse, M., Roberts, J.: Histogram matching and global initialization for laser-only slam in large unstructured environments. In: Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA 2007). Rome, Italy (2007)
11. Bosse, M., Zlot, R.: Map matching and data association for large-scale two-dimensional laser scan-based SLAM. Int. J. Robot. Res. **27**(6), 667–691 (2009)
12. Brscic, D., Hashimoto, H.: Model based robot localization using onboard and distributed laser range finders. In: Proceedings of the 2008 IEEE/RSJ International

Conference on Intelligent Robots and Systems (IROS 2008). Nice, France (2008)

13. Burgard, W., Trahanias, P., Hanel, D., Moors, M., Shulz, D., Baltzakis, H., Argyros, A.: Tele-presence in populated exhibitions through Web-operated mobile robots. Auton. Robots **15**, 299–316 (2003)
14. Choset, H., Nagatani, K.: Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization. IEEE Trans. Robot. Autom. **17**(2), 125–137 (2001)
15. Cox, I.: BLANCHE—an experiment in guidance and navigation of an autonomous robot vehicle. IEEE Trans. Robot. Autom. **7**(2), 193–204 (1991)
16. Dissanayake, G., Williams, S.B., Durrant-Whyte, H., Bailey, T.: Map management for efficient simultaneous localization and mapping (SLAM). Auton. Robots **12**, 267–286 (2002)
17. Doh, N., Lee, K., Chung, W., Cho, H.: Simultaneous localisation and mapping algorithm for topological maps with dynamics. IET Control Theory Appl. **3**(9), 1249–1260 (2009)
18. Dong, J., Wijesoma, S., Shacklock, A.: Extended Rao-Blackwellised genetic algorithmic filter SLAM in dynamic environments with raw sensor measurements. In: Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007). San Diego, CA (2007)
19. Duda, R., Hart, P.: Pattern Classification and Scene Analysis. Wiley and Sons, New York (1973)
20. Estrada, C., Neira, J., Tardós, J.: Hierarchical SLAM: real-time accurate mapping of large environments. IEEE Trans. Robot. **21**(4), 588–596 (2005)
21. Folkensson, J., Christensen, H.: Robust SLAM. In: Proceedings of the Fifth IFAC/EURON Symposium on Intelligent Autonomous Vehicles (IAV 2004). Lisbon, Portugal (2004)
22. Fox, D.: Markov localization: a probabilistic framework for mobile robot localization and navigation. Ph.D. thesis, Institute of Computer Science III, University of Bonn (1998)
23. Fox, D., Burgard, W., Dellaert, F., Thrun, S.: Monte Carlo localization: efficient position estimation for mobile robots. In: Proceedings of the 16th National Conference on Artificial Intelligence (NCAI 1999). Bhankrota, India (1999)
24. Fox, D., Burgard, W., Thrun, S.: Markov localization for mobile robots in dynamic environments. J. Artif. Intell. Res. **11**, 391–427 (1999)
25. Fox, D., Thrun, S., Burgard, W., Dellaert, F.: Particle Filters for mobile robot localization. In: Doucet, A., de Freitas, N., Gordon, N. (eds.) Sequential Monte Carlo Methods in Practice, Statistics for Engineering and Information Science. Springer, New York (2001)
26. Frese, U.: A discussion of simultaneous localization and mapping. Auton. Robots **20**, 25–42 (2006)
27. Georgiev, A., Allen, P.: Localization methods for a mobile robot in urban environments. IEEE Trans. Robot. **20**(5), 851–864 (2004)
28. Glas, D., Kanda, T., Ishiguro, H., Hagita, N.: Simultaneous people tracking and localization for social robots using external laser range finders. In: Proccedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009). Nice, France (2009)
29. Guivant, J., Nebot, E.: Optimization of the simultaneous localization and map-building algorithm for real-time implementation. IEEE Trans. Robot. Autom. **17**(3), 242–257 (2001)
30. Guivant, J., Nebot, E.: Solving computational and memory requirements of feature-based simultaneous localization and mapping algorithms. IEEE Trans. Robot. Autom. **19**(4), 749–755 (2003)
31. Haehnel, D., Fox, D., Burgard, W., Thrun, S.: A highly efficient FastSLAM algorithm for generating cyclic maps of large-scale environments from raw laser range measurements. In: Proceedings of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003). Las Vegas, USA (2003)
32. Haehnel, D., Triebel, R., Burgard, W., Thrun, S.: Map building with mobile robots in dynamic environments. In: Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA 2003). Taipei, Taiwan (2003)
33. Iagnemma, K., Kang, S., Shibly, H., Dubowsky, S.: Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers. IEEE Trans. Robot. **20**(5), 921–927 (2004)
34. Jensfelt, P., Christensen, H.: Pose tracking using laser scanning and minimalistic environmental models. IEEE Trans. Robot. Autom. **17**(2), 138–147 (2001)
35. Julier, S., Uhlman, J.: A counter example to the theory of simultaneous localisation and map building. In: Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA 2001). Seoul, Korea (2001)
36. Kaess, M., Dellaert, F.: A Markov Chain Monte Carlo approach to closing the loop in SLAM. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005). Barcelona, Spain (2005)
37. Konolige, K.: SLAM via variable reduction from constraint maps. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005). Barcelona, Spain (2005)
38. Konolige, K., Grisetti, G., Kummerle, R., Burgard, W., Limketkai, B., Vincent, R.: Efficient sparse pose adjustment for 2d mapping. In: Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010). Taipei, Taiwan (2010)
39. Lidoris, G., Wollherr, D., Buss, M.: Bayesian state estimation and behavior selection for autonomous robotic exploration in dynamic environments. In: Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008). Nice, France (2008)
40. Lisien, B., Morales, D., Silver, D., Kantor, G., Rekleitis, I., Choset, H.: The hierarchical atlas. IEEE Trans. Robot. **21**(3), 473–481 (2005)
41. Liu, Y., Emery, R., Chakrabati, D., Burgard, W., Thrun, S.: Using EM to learn 3D models of indoor environments with mobile robots. In: Proceedings of the

2001 International Conference on Machine Learning (ICML 2001). Berkshires, MA (2001)

42. Lopez, M., Bergasa, L., Barea, R., Escudero, M.: A navigation system for assistant robots using visually augmented POMDPs. Auton. Robots **19**, 67–87 (2005)
43. Martinelli, A.: The odometry error of a mobile robot with a synchronous drive system. IEEE Trans. Robot. Autom. **18**(3), 399–405 (2002)
44. Maybank, S.: Filter based estimates of depth. In: Proceedings of the 1990 British Machine Vision Conference (BMVC 1990), pp. 349–354. London, UK (1990)
45. Miller, I., Campbell, M.: Rao-Blackwellized particle filtering for mapping dynamic environments. In: Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA 2007). Rome, Italy (2007)
46. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B.: FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In: Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI 2003). Acapulco, Mexico (2003)
47. Neira, J., Tardós, J.: Data association in stochastic mapping using the joint compatibility test. IEEE Trans. Robot. Autom. **17**(6), 890–897 (2001)
48. Nguyen, V., Martinelli, A., Tomatis, N., Siegwart, R.: A comparison of line extraction algorithms using 2d laser rangefinder for indoor mobile robotics. In: Proceedings of the 2005 IEEE International Conference on Intelligent Robots and Systems (IROS 2005). Edmonton, Canada (2005)
49. Nieto, J., Bailey, T., Nebot, E.: Scan-SLAM: combining EKF-SLAM and scan correlation. In: Proceedings of the 2005 International Conference on Field and Service Robotics (FSR 2005). Port Douglas, Australia (2005)
50. Nieto, J., Guivant, J., Nebot, E.: DenseSLAM: the unidirectional information flow (UIF). In: Proceedings of the Fifth IFAC/EURON Symposium on Intelligent Autonomous Vehicles (IAV 2004). Lisbon, Portugal (2004)
51. Ojeda, L., Borenstein, J.: Methods for the reduction of odometry errors in over-constrained mobile robots. Auton. Robots **16**, 273–286 (2004)
52. Ojeda, L., Cruz, D., Reina, G., Borenstein, J.: Current-based slippage detection and odometry correction for mobile robots and planetary rovers. IEEE Trans. Robot. **22**(2), 366–378 (2006)
53. Olson, C.: Probabilistic self-localization for mobile robots. IEEE Trans. Robot. Autom. **16**(1), 55–66 (2000)
54. Pfister, S., Kreichbaum, K., Roumeliotis, S., Burdick, J.: Weighted range sensor matching algorithms for mobile robot displacement estimation. In: Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA 2002). Washington, DC (2002)
55. Pfister, S., Roumeliotis, S., Burdick, J.: Weighted line fitting algorithms for mobile robot map building and efficient data representation. In: Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA 2003). Taipei, Taiwan (2003)
56. Rady, S., Wagner, A., Badreddin, E.: Hierarchical localization using entropy-based feature map and triangulation techniques. In: Proceedings of the 2010 IEEE International Conference on Systems Man and Cybernetics (SMC 2010). Istanbul, Turkey (2010)
57. Ramos, F., Nieto, J., Durrant-Whyte, H.: Recognising and modelling landmarks to close loops in outdoor slam. In: Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA 2007). Rome, Italy (2007)
58. Sack, D., Burgard, W.: A comparison of methods for line extraction from range data. In: Proceedings of the 5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles (IAV 2004). Lisbon, Portugal (2004)
59. Scalmato, A., Sgorbissa, A., Capezio, F., Mastrogiovanni, F., Vernazza, P., Vernazza, T., Zaccaria, R.: Mobile robots in hospital environments: an installation case study. In: Proceedings of the 5th International Conference on Mobile Robots (ECMR 2011). Orebro, Sweden (2011)
60. Scheding, S., Dissanayake, G., Nebot, E., Durrant-Whyte, H.: An experiment in autonomous navigation of an underground mining vehicle. IEEE Trans. Robot. Autom. **15**(1), 85–95 (1999)
61. Schulz, D., Burgard, W., Fox, D., Cremers, A.: Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In: Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA 2001). Seoul, Korea (2001)
62. Sim, R., Roy, N.: Global A-optimal robot esploration in SLAM. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005). Barcelona, Spain (2005)
63. Simoncelli, M., Zunino, G., Christensen, H., Lange, K.: Autonomous pool cleaning: self localization and autonomous navigation for cleaning. Auton. Robots **9**, 261–270 (2000)
64. Stachniss, C., Grisetti, G., Burgard, W.: Recovering particle diversity in a Rao-Blackwellized particle filter for SLAM after actively closing loops. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005). Barcelona, Spain (2005)
65. Thrun, S., Beetz, M., Bennewitz, M., Burgard, W., Creemers, A., Dellaert, F., Fox, D., Hahnel, D., Rosenberg, C., Roy, N., Schulte, J., Schulz, D.: Probabilistic algorithms and the interactive museum tour-guide robot Minerva. Int. J. Rob. Res. **19**(11), 972–999 (2000)
66. Thrun, S., Martin, C., Liu, Y., Hahnel, D., Emery-Montemerlo, R., Chakrabarti, D., Burgard, W.: A real-time Expectation-Maximization algorithm for acquiring multilanar maps of indoor environments with mobile robots. IEEE Trans. Robot. **20**(3), 433–442 (2004)
67. Thrun, S., Montemerlo, M., Koller, D., Wegbreit, B., Nieto, J., Nebot, E.: FastSLAM: an efficient solution to the simultaneous localization and mapping problem with unknown data association. J. Mach. Learn. Res. **4**, 1–48 (2004)
68. Tur, J., Gordillo, J., Borja, C.: A closed-form expression for the uncertainty in odometry position estimate

of an autonomous vehicle. IEEE Trans. Robot. **21**(5), 1017–1022 (2005)

69. Vu, T.D., Burlet, J., Aycard, O.: Grid-based localization and online mapping with moving objects detection and tracking: new results. In: Proceedings of the 2008 IEEE Intelligent Vehicles Symposium (IV 2008). Eindhoven, Netherlands (2008)
70. Wolf, D., Sukhatme, G.: Online simultaneous localization and mapping in dynamic environments. In: Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA 2004). New Orleans, LA (2004)
71. Wolf, D., Sukhatme, G.: Mobile robot simultaneous localization and mapping in dynamic environments. Auton. Robots **19**, 53–65 (2005)
72. Wong, R., Xiao, J., Joseph, S.: A robust data association for simultaneous localization and mapping in dynamic environments. In: Proceedings of the 2010 IEEE International Conference on Information and Automation (ICIA 2010). Heilongjiang, China (2010)
73. Wulf, O., Lecking, D., Wagner, B.: Robust self-localization in industrial environments based on 3d ceiling structures. In: Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006). Beijing, China (2006)
74. Zhou, H., Sakane, S.: Localizing objects during robot slam in semi-dynamic environments. In: Proceedings of the 2008 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2008). Xian, China (2008)