Auton Robot
DOI 10.1007/s10514-009-9108-0

On the use of likelihood fields to perform sonar scan matching localization

Antoni Burguera · Yolanda González · Gabriel Oliver

Received: 8 May 2008 / Accepted: 6 January 2009

Abstract Scan matching algorithms have been extensively used in the last years to perform mobile robot localization. Although these algorithms require dense and accurate sets of readings with which to work, such as the ones provided by laser range finders, different studies have shown that scan matching localization is also possible with sonar sensors. Both sonar and laser scan matching algorithms are usually based on the ideas introduced in the ICP (*Iterative Closest Point*) approach. In this paper a different approach to scan matching, the Likelihood Field based approach, is presented. Three scan matching algorithms based on this concept, the non filtered sNDT (*sonar Normal Distributions Transform*), the filtered sNDT and the LF/SoG (*Likelihood Field/Sum of Gaussians*), are introduced and analyzed. These algorithms are experimentally evaluated and compared to previously existing ICP-based algorithms. The obtained results suggest that the Likelihood Field based approach compares favorably with algorithms from the ICP family in terms of robustness and accuracy. The convergence speed, as well as the time requirements, are also experimentally evaluated and discussed.

Keywords Sonar · Scan matching · Likelihood fields

This work is partially supported by DPI 2005-09001-C03-02 and FEDER funding.

A. Burguera (✉) · Y. González · G. Oliver
Dept. Matemàtiques i Informàtica, Universitat de les Illes Balears, Ctra. Valldemossa, Km. 7'5, 07122 Palma de Mallorca, Illes Balears, Spain
e-mail: antoni.burguera@uib.es

Y. González
e-mail: y.gonzalez@uib.es

G. Oliver
e-mail: goliver@uib.es

1 Introduction

Nowadays an outstanding issue in robotics is mobile robot localization. Thrun et al. (2005) define mobile robot localization as the problem of determining the pose of a robot relative to a given map of the environment. However, in many robotic applications it is not possible to have an *a priori* map of the environment. In such situations, the problem may be addressed by building local maps of the environment while the robot is executing a mission and, subsequently, determining the robot pose by matching the local maps.

The choice of a specific map to represent the knowledge regarding the environment is a difficult task. Maps can be roughly classified in two categories, referred to as *Feature-Based Maps* and *Location-Based Maps*.

Feature-based maps, which are composed of a set of features together with their Cartesian location, are extensively used in the localization context. However, these maps introduce geometric constraints such as the existence of lines and corners in the environment (Castellanos et al. 2001; Dissanayake et al. 2002; Bosse et al. 2004). Thus, they are not well suited to model non-structured environments.

Location-based maps, which offer a marker for any location in the world, usually do not assume geometric constraints. A classical location-based map is the *occupancy grid* (Moravec 1988; Elfes 1989). These maps have two major drawbacks to performing localization. First, it is computationally expensive to match such dense representations of the space. Second, the granularity inherent to a grid representation may produce low resolution estimates of the robot pose.

To deal with those problems, some authors determine the robot displacement by matching up successive sets of raw range readings, called *scans*. This technique is known as *scan matching* (Burguera et al. 2005; Pfister et al. 2002;

Published online: 29 January 2009

Weiss and Puttkamer 1995). Scan matching neither assumes geometric constraints nor builds dense, grid-based, representations of the environment. Thus, it is well suited to localizing a mobile robot with high precision both in structured and non-structured environments.

First attempts to perform mobile robot localization by matching successive range scans were inspired by the computer vision community. A standard approach to image registration is the ICP (*Iterative Closest Point*). Although the name ICP was first presented by Besl and McKay (Besl and McKay 1992), very similar ideas were presented by other authors such as Chen and Medioni (1992) or Zhang (1994).

The ICP concepts were introduced in the mobile robot localization context by Lu and Milios (1997). These authors proposed some changes to the original algorithm to make it more suitable for robotic applications. As a matter of fact, due to the great success of this approach, many other scan matching algorithms rely on the same basic structure. In the context of this paper, these algorithms will be referred to as *ICP-based* algorithms.

Although different localization strategies have been used with a variety of range sensors, such as sonar (Leonard and Durrant-Whyte 1992; Tardós et al. 2002), ICP-based algorithms have been mostly used to localize mobile robots endowed with laser range finders. Today, off the shelf laser sensors provide thousands of readings per second with a sub degree angular resolution. Other sensors, such as standard Polaroid ultrasonic range finders, are only able to provide tenths of readings per second, with angular resolutions one or two orders of magnitude poorer than laser. Moreover, effects such as multiple reflections or cross-talking are very frequent in sonar sensing, producing large amounts of readings which do not correspond to real objects in the environment.

However, ultrasonic range finders have interesting properties that make them appealing in the mobile robotics community (Lee 1996). On the one hand, their size, power consumption and price are better than those of laser scanners. Consequently, they are well suited for low cost and domestic robots, such as automatic vacuum cleaners. On the other hand, their basic behavior is shared by underwater sonar sensors, which are extensively used in underwater and marine robotics. Thus, typical underwater sonar, although being far more complex than standard Polaroid ultrasonic range finders, can benefit from of those localization techniques that take into account the sonar limitations.

Recent studies demonstrated that ICP-based algorithms can be used with sonar sensors (Burguera et al. 2005), especially if an accurate sensor model is defined (Burguera et al. 2007c). For instance, in (Burguera et al. 2008), each sonar reading is modeled as a Normal distribution, thus accounting for range and angular uncertainties. Then, statistical compatibility tests are used to establish reading to reading correspondences in an ICP-based framework.

In spite of its great success, ICP-based algorithms present important flaws. Other paradigms have been recently introduced to avoid such problems. For example, the NDT (*Normal Distributions Transform*) (Biber and Straßer 2003) defines the scan matching concept from a completely different point of view, which will be described in detail later in this document. Also, Hähnel et al. (2003) define a probabilistic laser scan registration. In this study, hill-climbing strategies are used to maximize a likelihood function built from a laser range finder model. In order to evaluate this model, ray-tracing techniques have to be applied. These new paradigms are strongly related to the concept of *Likelihood Field*, which is a common and computationally cheap way to model range sensors and perform localization given an *a priori* map. Nevertheless, these paradigms rely on accurate and dense sets of laser range readings, and are not suited to working with sonar sensors.

Our objective is to define new sonar scan matching paradigms which do not rely on the establishment of correspondences. This paper focuses on the use of Likelihood Fields to perform such a task. Henceforth, the scan matching techniques relying on Likelihood Fields will be referred to as Likelihood Field based or *LF-based* for short. The theoretical basis of LF-based scan matching is presented, and two new variants of Likelihood Fields to be used with sonar readings are introduced.

The main contributions of this paper are as follows; first, the theoretical basis of the LF-based scan matching is provided. This theoretical basis is a generalization of the one presented by Biber et al. (2004), where Likelihood Fields were not explicitly taken into account. Then, two novel methods to achieve robust and accurate scan matching localization, particularly suitable when sonar sensors are used, are introduced. At first, the sNDT (*sonar Normal Distributions Transform*), which involves important arrangements to the original NDT concept to cope with the high number of outliers provided by sonar sensors. Secondly, the LF/SoG (*Likelihood Field/Sum of Gaussians*), which is originally designed to work with sonar sensors but can be directly used with other range sensors. Both methods are evaluated by means of a complete set of experiments, comparing them with other well known scan matching algorithms.

Although not being the central point of this paper, the required processes to deal with the sparsity of sonar readings are also presented as a necessary tool both for sNDT and LF/SoG.

2 Notation

Scan matching algorithms require two sets of range readings called *scans*. Let $S_{ref} = \{q_1, q_2, \ldots, q_n\}$ be a set of n points gathered at frame A, which is called the *reference scan*. Let $S_{cur} = \{p_1, p_2, \ldots, p_m\}$ be a set of m points gathered at frame B, which is called the *current scan*.

The aim of scan matching algorithms is to estimate the robot motion between frames A and B. Let x_B^A be the scan matching estimate of the relative position between the frames A and B. Thus, x_B^A represents a rototranslation in the plane. A common assumption is that the error in scan matching is Normal. Thus, we will represent the scan matching estimate as a multivariate Normal distribution $x_B^A = N(\hat{x}_B^A, P_B^A)$. Being x_B^A a rototranslation in the plane, the mean vector $\hat{x}_B^A$ has the form $[x, y, \theta]^T$, where x and y represents the translation and θ represents the rotation. Accordingly, the covariance P_B^A is a 3×3 matrix.

To deal with processes corrupted by Gaussian noise, a common approach in stochastic mapping and SLAM is the use of the operator $\oplus$ (composition of transformations). This operator is used throughout this paper both for points and transformations. A detailed description can be found in (Tardós et al. 2002).

Let $S'_{cur} = \{p'_1, p'_2, \ldots, p'_m\}$ be the set of S_{cur} points projected to the coordinate frame of S_{ref}. That is, $p'_i = x_B^A \oplus p_i$, $\forall p_i \in S_{cur}$.

Broadly speaking, the problem of scan matching is one of finding the robot motion x_B^A that maximizes the overlap between the parts of the environment represented by S_{ref} and S_{cur}. The intuitive concept of overlap turns into very different mathematical definitions depending on the specific scan matching algorithm under consideration. For instance, in the ICP context, the overlap is a function of the sum of distances between pairs of closest points in S_{ref} and S'_{cur}. In the Likelihood Field based approach, the overlap is represented by the likelihood of having the readings in S'_{cur} given a likelihood function constructed according to S_{ref}. The formal definitions of overlap for the different approaches to scan matching presented in this paper are provided in Sects. 3 and 5. Finally, Table 1 summarizes the main acronyms used throughout this paper.

Table 1 Table of acronyms

Acronym	Name
ICP, sICP	(sonar) Iterative Closest Point
IDC, sIDC	(sonar) Iterative Dual Correspondence
MbICP	Metric Based Iterative Closest Point
pIC, spIC	(sonar) Probabilistic Iterative Correspondence
NDT, sNDT	(sonar) Normal Distributions Transform
LF/SoG	Likelihood Field/Sum of Gaussians

3 ICP-based algorithms

A general description of ICP-based scan matching algorithms is provided next so that their behavior and main drawbacks can be understood and, later, compared to our proposal, the LF-based approach.

3.1 Overview

ICP-based algorithms consist of an iterative process to estimate the robot displacement and rotation that maximize the overlap between two consecutive sensor scans by establishing correspondences involving points in the two scans.

Let $x_{B_k}^A = [x_k, y_k, \theta_k]^T$ be the ICP-based scan matching estimate of the relative position between the frames A and B at the iteration k, k being 0 at the beginning of the process. A particular case is, thus, $x_{B_0}^A$, which is the initial estimate. This initial estimate has to be computed by other means, for instance, using odometry.

In an ICP-based algorithm, at iteration $k > 0$, the following three steps are executed until convergence is achieved.

- S_{cur} is expressed with respect to frame A using $x_{B_{k-1}}^A$. Let $S'_{cur_k} = \{p'_1, p'_2, \ldots, p'_m\}$ be the set of transformed points. That is $p'_i = x_{B_{k-1}}^A \oplus p_i, \forall p_i \in S_{cur}$.
- For each point $q_i \in S_{ref}$, the corresponding point p'_j is determined as the closest point in S'_{cur_k} whose distance $d(q_i, p'_j)$ is below a certain threshold d_{min}. The set of correspondences $C_k = \{(i, j) \mid 1 \leq i \leq n, 1 \leq j \leq m\}$ is obtained.
- The motion estimate $x_{B_k}^A$ that minimizes the sum of squared distances between q_i and $x_{B_k}^A \oplus p_j$ is computed, being $(i, j) \in C_k$.

When convergence is achieved, the algorithm ends and $x_{B_k}^A$ constitutes the solution x_B^A. Otherwise, the process iterates.

As stated previously, the goal of scan matching is to find the x_B^A that maximizes the overlap between S_{ref} and S_{cur}. In the last step of the described ICP-based algorithm, a function of S_{ref}, S_{cur} and x_B^A is minimized. Accordingly, the overlap is measured, in the ICP context, as minus the previous function. Obviously, the goal of the minus sign is to turn the problem of maximizing the overlap into one of minimizing the sum of squared distances.

The main difference between ICP-based algorithms is how the distance $d(q_i, p'_j)$ between scan points is computed. The original ICP uses Euclidean distance. Although the use of this distance provides good estimates of the displacement between the scans, it does not do so when dealing with rotation. Among others, IDC (*Iterative Dual Correspondence*) and MbICP (*Metric-based Iterative Closest Point*) (Minguez et al. 2006) deal with the rotation problem. IDC establishes two sets of correspondences; one dealing with the translation

using Euclidean distance and the other with the rotation by means of an angular distance. MbICP defines a new distance measure that simultaneously accounts for translation and rotation errors. However, none of these methods take into account the range sensor imprecisions. Some other methods, such as the research of Pfister et al. (2002) and the pIC (*probabilistic Iterative Correspondence*) (Montesano et al. 2005) deal with this problem. The former, by weighting the contribution of each scan point according to its uncertainty. The latter defines an interesting framework to deal with uncertainties in scan matching computing statistical compatibility between the scan points by means of the Mahalanobis distance.

3.2 Drawbacks

In spite of its high success, ICP-based algorithms have two important drawbacks, which are described next.

Correspondences: What scan matching wants to estimate is a roto-translation between two nearby places in the environment. However, this estimation is carried out by means of the partial and noisy projections of the environment provided by range sensors. This fact is particularly relevant when observing the concept of correspondence. When point-to-point correspondences are established it is implicitly assumed that corresponding points have been produced *exactly* at the same position in the environment. This assumption is not correct, not only because of the existence of spurious readings, but mainly because sensors *sample* the environment, providing a discrete view of its surroundings. Moreover, at a given iteration k, only points belonging to C_k are taken into account to compute $x^A_{B_k}$. Thus, the problem is not only that sensors provide partial views of the environment, but also that the algorithm itself discards information when establishing correspondences. Different attempts have been performed to minimize these effects, ranging from point-to-line correspondences (Lu and Milios 1997) to the use of statistical compatibility tests (Montesano et al. 2005; Burguera et al. 2008). However, the problems derived from the establishment of correspondences are inherent to the ICP-based approach. Thus, to solve them, a different scan matching approach is necessary.

Poor convergence: ICP-based algorithms do not hold information to guarantee coherence between iterations. Once a minimization is performed, a new set of correspondences is established. Moreover, the threshold d_{min} is context dependent and has to be tuned. Depending on the value of this parameter, the algorithm can discard too many readings when searching corresponding points. In these cases, the set of correspondences changes a lot from one iteration to the next. Thus, the function to be minimized is changing in every iteration due to the different correspondence sets. Depending on the threshold d_{min} and, thus, on the specific context where the ICP-based algorithm is deployed, the changes on the function being minimized may result in a large number of iterations until the algorithm converges. In some cases, changing correspondences may result in oscillations around a minimum and even prevent convergence.

In order to deal with these problems, new approaches to scan matching are necessary. Although the concept of Likelihood Field is not new, its use in the scan matching context prevents the establishment of correspondences and, thus, solves the two aforementioned problems. In consequence, the use of Likelihood Fields in scan matching seems to be a good approach to overcome some of the ICP-based limitations.

4 Sonar scan matching

Figure 1 summarizes the necessary processes to perform sonar scan matching. The *matching* box is in charge of finding the robot motion that maximizes the overlap between the two scans. In other words, it corresponds to the same concept as scan matching using laser sensors. The rest of the processes, which are necessary to tackle the sparsity of sonar readings, are outlined next. Only short descriptions are provided to give an overview of the full sonar scan matching process. Detailed descriptions are beyond the scope of this paper. More details can be found in (Burguera et al. 2008)

4.1 Measurement grouping

The scan concept itself is not of direct application to ultrasonic range finders. Instead of having a single sensor scanning the environment, as in the case of a laser range finder, a set of sonars at fixed poses with respect to the robot are used. Thus, a whole view of the environment consists only of a few readings, usually between eight and twenty four, depending on the robot configuration.

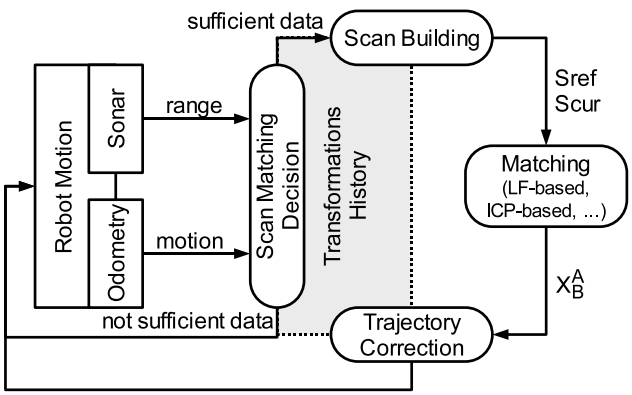

Fig. 1 Overview of the sonar scan matching processes

Scan matching algorithms require dense sets of readings with which to work. Accordingly, to perform sonar scan matching, some processes to group the sonar readings along short robot trajectories are needed. These processes will be referred to as the *Scan Matching Decision* and the *Scan Building*.

The Scan Matching Decision is in charge of collecting sonar readings and odometric estimates and storing them in the so called *Transformations History*. The Scan Matching Decision is also in charge of deciding when sufficient data has been collected so that a scan can be built. When sufficient sensor readings have been collected, according to the Scan Matching Decision, the Scan Building process is executed. This process makes use of the Transformations History and builds the sonar scan by representing the stored sonar readings with respect to a common coordinate frame.

When the two scans S_{cur} and S_{ref} have been generated by the Scan Building, the matching process (either ICP-based or LF-based) is executed.

4.2 Trajectory correction

After execution of the matching process, the estimate x_B^A between the coordinate frames of S_{ref} and S_{cur} is available. However, the scan readings have not only been gathered at frames A and B, but along a much larger set of robot poses. Thus, it is desirable to correct all the motion estimates involved in the Scan Building process according to x_B^A. This task is performed by the *Trajectory Correction* process.

The Trajectory Correction computes the most probable trajectory followed by the robot that agrees with the estimation provided by the scan matching. This can be expressed as a constrained optimization problem. In (Burguera et al. 2008) a method based on an IEKF (*Iterated Extended Kalman Filter*) is proposed to solve this problem. Figure 2 shows an example of the effect of the Trajectory Correction process.

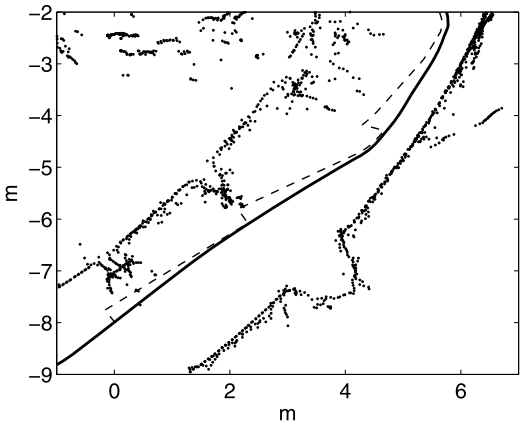

Fig. 2 Example of trajectory correction. The *line of dashes* shows the trajectory prior to the trajectory correction. The *continuous line* shows the corrected trajectory

5 LF-based algorithms

This section focuses on the definition of Likelihood Fields as a framework to register point clouds in the mobile robot localization context. Biber et al. (2004) derive the theory behind the NDT from a probabilistic interpretation of the ICP algorithm. They formulate the estimation of the robot motion as a Maximum Likelihood problem and propose Newton's algorithm as a way to perform the optimization. In Sect. 5.1, these concepts are detached from the NDT and ICP algorithms and formulated in the more general context of Likelihood Fields, which are also defined. In Sect. 5.2, Newton's algorithm is clearly stated and linked to the proposed Likelihood Field approach. Additionally, Biber (2007) used some properties of Newton's algorithm in the NDT context to compute the error of the scan matching estimate in form of covariance matrix. They made some assumptions that are not realistic if noisy sets of sonar readings are used. Section 5.3 extends these concepts to the more general Likelihood Field context and adapts them, thus rendering them suitable in the sonar context too.

The theory provided in this section constitutes the basis of two algorithms, the sNDT and the LF/SoG that will be described later in this paper.

5.1 Overview

A Likelihood Field is defined (Thrun 2001) as a function of $x-y$-coordinates depicting the likelihood of obstacle detection. Likelihood Fields are usually built using an *a priori* map.

In the context of this paper, a Likelihood Field can be defined as follows. A Likelihood Field is a function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ computing the likelihood of having a range reading at given $x-y$-coordinates. Moreover, this function is constructed using the range readings in S_{ref}, and not *a priori* maps.

It is important to emphasize that a Likelihood Field is not a PDF (*Probability Density Function*). However, turning a Likelihood Field into a PDF only involves the use of a normalization factor η. To fully understand the utility of Likelihood Fields in scan matching, let us define the PDF $g(x)$ as follows:

$$g(x) = \eta f(x). \tag{1}$$

The underlying idea is to consider $g(x)$ as a *generative process* for S'_{cur}: it is assumed that a point $p'_i \in S'_{cur}$ has been generated by drawing from the probability distribution defined by $g(x)$. Being S_{cur} the result of composing x_B^A

with each point in S_{cur}, the problem of scan matching can be seen as the one of finding the x_B^A that makes the generative process assumption true. From a probabilistic point of view, this can be expressed as the problem of maximizing the following likelihood function:

$$\Psi(x) = \prod_{p_i \in S_{cur}} g(x \oplus p_i) \tag{2}$$

where x represents a rototranslation between the Likelihood Field and the S_{cur} coordinate frames. The idea behind this function is to project each point in the current scan onto the PDF g by means of the aforementioned rototranslation. Then, the PDF is evaluated at each of these projected points and the results are multiplied. A good rototranslation x will project the points in S_{cur} onto regions of g with high values (i.e. with high probability of having a sonar reading). Thus, the better the rototranslation x, the higher the values of $\Psi(x)$.

The rototranslation x that maximizes this likelihood function constitutes the scan matching estimate x_B^A. In consequence, the likelihood function $\Psi(x)$ represents the overlap between the two scans. If the NDT grid is used as the Likelihood Field $f(x)$, the likelihood function $\Psi(x)$ turns into the energy function proposed in (Biber et al. 2004).

As it may be computationally expensive to maximize Eq. 2, a usual approach is to use log-likelihood functions. Thus, the problem of scan matching using Likelihood Fields can be expressed as one of minimizing the following negative log-likelihood function:

$$\begin{aligned} &-\log(\Psi(x)) \\ &\quad = -\sum_{p_i \in S_{cur}} \log(g(x \oplus p_i)) \\ &\quad = -\left((m \log \eta) + \sum_{p_i \in S_{cur}} \log(f(x \oplus p_i)) \right) \end{aligned} \tag{3}$$

where m is the number of points in S_{cur}. η being a constant value, it is clear that it will not influence the minimization process. Thus, it is not necessary either to compute it or to turn the Likelihood Field into a PDF.

This approach has some advantages when compared to the ICP-based approach. On the one hand, no correspondences are established. On the other hand, only one function has to be minimized. This is important because, in the ICP-based context, since the set of correspondences changes at every iteration, then so too the function to be minimized. Thus, although the ICP-based structure is the same throughout its whole execution, it minimizes a different function at every iteration. Some problems arise from this issue, as described in Sect. 3.2. Moreover, the underlying probabilistic approach to LF-based scan matching makes it easy to estimate the error of the matching process in the form of covariance matrix, as will be shown later.

The key issue when using Likelihood Fields is how to build an accurate Likelihood Field from sensor data. The main difference between the approaches presented in this document refers to how the Likelihood Field is built.

5.2 Optimization

The optimization process consists of minimizing Eq. 3. As stated previously, the constant term $m \log \eta$ does not affect the minimization. Thus, to provide a simpler notation, let the function to be minimized be as follows:

$$h(x) = -\sum_{p_i \in S_{cur}} \log(f(x \oplus p_i)). \tag{4}$$

Newton's algorithm has proved to be an effective tool in this context (Biber and Straßer 2003), and has interesting properties that will be shown later. Newton's algorithm is commonly used to find the minima by using the gradient vector and the Hessian matrix instead of the function itself and the gradient vector. Next, Newton's method for finding a minimum of the score function is presented.

1. Start with an approximation x_0 to the minimum point. This approximation can be obtained using odometry. Set $k \leftarrow 0$.
2. Evaluate the gradient vector $\nabla h(x_k)$ and the Hessian matrix $H(x_k)$.
3. Compute the next estimate $x_{k+1} \leftarrow x_k + \Delta x$, being $\Delta x = -(H(x_k))^{-1} \nabla h(x_k)$
4. If convergence is achieved, the algorithm ends and x_{k+1} constitutes the scan matching estimate x_B^A. Otherwise, set $k \leftarrow k + 1$ and iterate.

Newton's algorithm to find the minima of a function is proposed in this paper to perform the optimization in the context of LF-based scan matching. Other minimization algorithms could be used and, of course, if a closed form solution exists for one specific Likelihood Field definition, is greatly preferred.

5.3 Estimating the covariance matrix

In (Biber 2007) some properties of Newton's algorithm were used to estimate the error of the scan matching estimate. This section outlines the mentioned approach and shows how this approximation of the scan matching error is valid in other Likelihood Field approaches providing Newton's algorithm is used to perform the optimization.

At each iteration, Newton's algorithm approximates the function being minimized, $h(x)$, by a quadratic. In particular, when convergence is achieved at the minimum x_B^A, this quadratic has the form:

$$h(x) \simeq h(x_B^A) + \frac{1}{2}(x - x_B^A)^T H(x_B^A)(x - x_B^A). \tag{5}$$

A common approach is to model the scan matching error as a Normal distribution with mean x_B^A and covariance P_B^A. Thus, the PDF of the error is the following Gaussian:

$$p_{error}(x) = \eta' \exp\left(-\frac{1}{2}(x - x_B^A)^T (P_B^A)^{-1}(x - x_B^A)\right) \quad (6)$$

The negative log-likelihood of $p_{error}(x)$ is as follows:

$$-\log(p_{error}(x)) = -\log(\eta') + \frac{1}{2}(x - x_B^A)^T (P_B^A)^{-1}(x - x_B^A). \quad (7)$$

It has implicitly been assumed that the quadratic in Eq. 5 is a sufficiently good approximation of $h(x)$ when Newton's algorithm has converged to a minimum. In other words, it is assumed that the higher order terms of the Taylor series are small. It can be observed in Eq. 4 that $h(x)$ is constructed by projecting the points in S_{cur} onto the Likelihood Field. Thus, it is assumed that the range readings are properly modeled by points. This is a reasonable assumption if laser sensors are used. Due to the very high accuracy of laser, the readings can be assumed to be points truly located at the detected object. In consequence, the function $h(x)$ will most likely have a clear minimum around the true robot motion and Newton's algorithm may converge to a x_B^A very close to that minimum.

On the contrary, if sonar sensors are used, the point reading assumption is not as good as it was with laser. Mainly because of its low angular resolution, an object detected by an ultrasonic range finder may not be located on the point reading, but in a certain area around the point. Thus, some ambiguities may appear, resulting in a minimum of $h(x)$ not being as clear as in the case of laser sensors. Thus, Newton's algorithm may not be able to converge to a solution x_B^A as close to the minimum as in the case of laser.

As stated previously, Eq. 5 approximates $h(x)$ only around a minimum. In other words, the closer x_B^A is to the real minimum, the better the approximation. As the research in (Biber 2007) is based on laser sensors, it is assumed that Eq. 5 constitutes a very good approximation of $h(x)$. In consequence, taking into account Eqs. 5 and 7, and performing some additional assumptions, they conclude that $P_B^A = H(x_B^A)^{-1}$. In other words, the scan matching error can be easily computed from the Hessian matrix used by Newton's algorithm.

However, in the sonar case, although the x_B^A computed by Newton's algorithm may be good enough to localize the robot, it can not guarantee that Eq. 5 is an approximation of $h(x)$ as good as that of the laser case. In consequence, in the sonar case we can not conclude that the scan matching error is equal to $H(x_B^A)^{-1}$. Nevertheless, looking at Eqs. 5 and 7 it is clear that the scan matching error, although being bigger in the sonar case, has a similar shape to the one of the laser case. At this point, we have experimentally observed

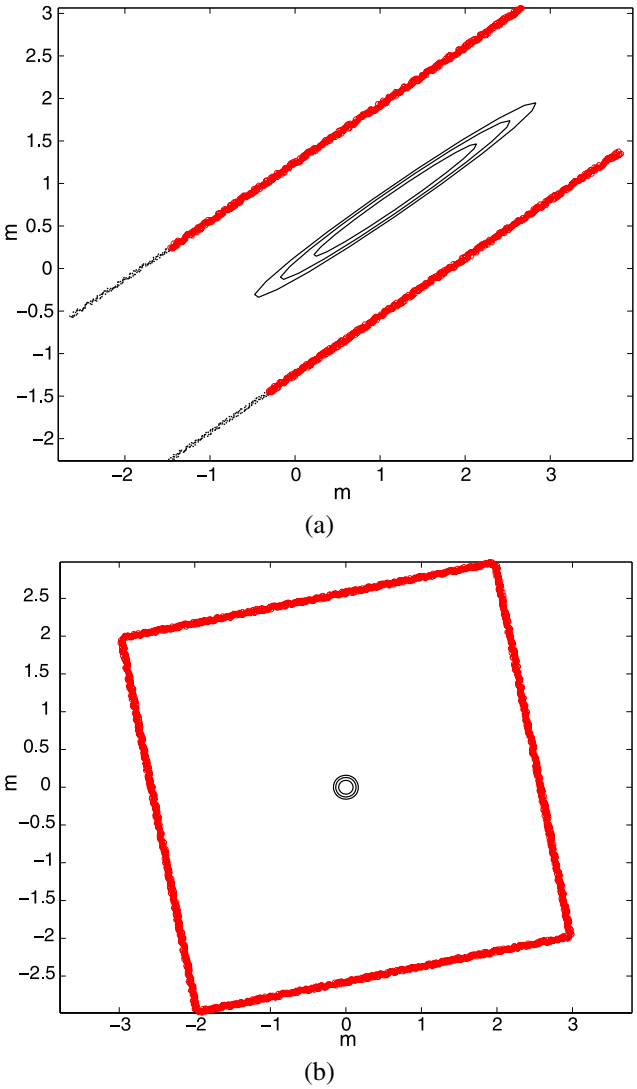

Fig. 3 Example of covariance matrices of the error when performing the matching in (**a**) a synthetic straight corridor and (**b**) a synthetic square room. The 99% confidence *ellipses* drawn correspond to $K = 10^4$, for the inner one, $K = 2 \times 10^4$, for the central *ellipse*, and $K = 3 \times 10^4$ for the external one

that the lower sonar accuracy can be accounted for, when computing the scan matching error, by assuming that P_B^A is proportional to $H(x_B^A)^{-1}$

$$P_B^A = H(x_B^A)^{-1} K. \quad (8)$$

The problem is, now, how to find the value of K. Its value can not be deduced from previous equations as it depends on the specific function being optimized. Thus, it should be tuned experimentally for each LF-based scan matching method.

Figure 3 depict the 99% confidence ellipses for different values of K. The scans used in the matching process are synthetic to provide two clear examples of a perfect corridor and a square room. It can be observed how, in Fig. 3a the el-

lipse spreads out along the corridor, which is the direction of maximum uncertainty. In Fig. 3b the uncertainty region has a circular shape, as in the square room there is no predominant direction for the error. Moreover, the ellipses in Fig. 3b are much smaller than those in Fig. 3a because the matching was more accurate.

6 The sonar normal distributions transform (sNDT)

The sNDT constitutes a good example of the LF-based sonar scan matching. Being an improvement of the NDT (Biber and Straßer 2003) to deal with sonar sensors, it demonstrates how an existing scan matching algorithm can be used with sonar sensors if the real behavior of these sensors is explicitly taken into account.

The main structure of the sNDT algorithm coincides with the NDT. First, a grid is built. The idea behind this grid is similar to the one of Likelihood Field. When the grid has been built, a score function is defined so that optimizing this function leads to the solution of the scan matching. The sNDT structure is the same, except that the grid building process is different and that a filtering procedure is applied to the scans. For this reason, the NDT approach is first presented, though adapted slightly to the Likelihood Field background. Then, the two improvements that define the sNDT are presented.

6.1 The normal distributions transform (NDT)

Biber and Straßer presented the NDT approach in (Biber and Straßer 2003) as an *ad hoc* method to register point clouds. The idea was refined and a probabilistic interpretation was provided in (Biber et al. 2004). The key idea of the Normal Distributions Transform (NDT) is to model the distribution of a point cloud by a grid of Normal distributions. A description of the original NDT is provided in this section, albeit adapted slightly to the approach presented in this paper; the LF-based scan matching.

The NDT grid is built from $q_i \in S_{ref}$. It starts dividing the space containing S_{ref} in N cells of size $L \times L$ and searching the set of q_i points lying inside each cell. In the seminal NDT paper, the value of L was set to 1m. Depending on the position of the grid's origin in the range $[0, L) \times [0, L)$, the points of S_{ref} will be divided in different groups. Let us denote by α a particular position of the grid's origin. Then, for each cell j containing at least three points, the following steps are executed:

1. Let $\Omega_{\alpha,j}$ be the set of n points $q_i \in S_{ref}$ contained in this cell.
2. Compute the mean $\mu_{\alpha,j}$ and the covariance matrix $P_{\alpha,j}$ of the points in $\Omega_{\alpha,j}$.
3. To prevent singular and near singular covariance matrices, the smallest eigenvalue of $P_{\alpha,j}$ is tested to be at least 0.001 times the biggest eigenvalue. If not, it is set to this value. The parameter 0.001 was experimentally tuned in (Biber and Straßer 2003).
4. Model the probability of having a reading at point x contained in cell j by the bivariate Normal distribution $N(\mu_{\alpha,j}, P_{\alpha,j})$. By dropping the normalization factor off the PDF (see Eq. 1), the Likelihood Field corresponding to cell j is as follows:

$$f_{\alpha,j}(x) = \exp\left(-\frac{(x-\mu_{\alpha,j})^T P_{\alpha,j}^{-1}(x-\mu_{\alpha,j})}{2}\right). \tag{9}$$

The Likelihood Field $f_\alpha(x)$ associated to S_{ref} and the grid's origin α is then built using the computed $f_{\alpha,j}(x)$ as follows:

$$f_\alpha(x) = \begin{cases} f_{\alpha,1}(x), & x \in \Omega_{\alpha,1}, \\ f_{\alpha,2}(x), & x \in \Omega_{\alpha,2}, \\ \dots & \\ f_{\alpha,N}(x) & x \in \Omega_{\alpha,N}. \end{cases} \tag{10}$$

As a result of the previous algorithm, each cell in the space containing S_{ref} is modeled by a bivariate Normal distribution. This may be a good approximation locally, however, important discontinuities appear in the cell limits. Figure 4a illustrates this point. Using this grid may lead to problems similar to those of establishing correspondences in the

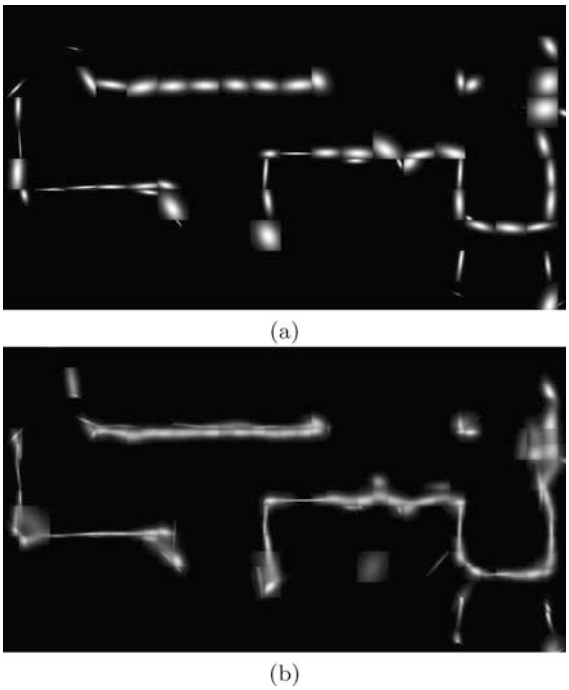

Fig. 4 Likelihood Fields generated by NDT being $L = 1$ m. The represented area is 18 m×10 m (**a**) Single grid. (**b**) Sum of four overlapping grids

ICP-based approach. Moreover, the minimization algorithm requires the Likelihood Field to be continuous and differentiable and certainly this is not true in that model.

To reduce the effects of these problems, the original NDT approach proposes the use of four overlapping grids, instead of a single one. Now, each point falls into four cells. Thus, four Likelihood Fields are built, each of them considering a particular position of the grid's origin. Given a point x, four likelihood values, $f_1(x)$, $f_2(x)$, $f_3(x)$ and $f_4(x)$, are available. Using this approach, to evaluate the Likelihood Field at position x, the contributions of the four grids are added up

$$f(x) = \sum_{1 \leq \alpha \leq 4} f_\alpha(x). \tag{11}$$

Figure 4b shows the result of summing the four overlapping grids. Although the function is not continuous, it is, in practice, usable in Newton's algorithm context.

When $f(x)$ is defined, the optimization process has to be carried out. According to Eq. 4, the function to be optimized in the NDT approach should be as follows:

$$h(x) = -\sum_{p_i \in S_{cur}} \log\left(\sum_{1 \leq \alpha \leq 4} f_\alpha(x \oplus p_i)\right). \tag{12}$$

However, instead of minimizing $h(x)$ as defined in the previous Equation, the NDT minimizes the following score function, $s(x)$:

$$s(x) = -\sum_{p_i \in S_{cur}} \sum_{1 \leq \alpha \leq 4} f_\alpha(x \oplus p_i). \tag{13}$$

The main advantage of defining this score function instead of the negative log-likelihood is that it makes the optimization process easier and faster. Moreover, (Biber et al. 2004) shows that $s(x)$ is a good approximation of $h(x)$. Thus, the NDT computes x_B^A by minimizing the score function $s(x)$ by means of Newton's algorithm as described in Sect. 5.2.

6.2 Building the sNDT grid

As stated previously, the sNDT follows the same structure as the NDT except that the grid is built taking into account the sonar behavior and the scans are filtered. Next, both the sNDT grid building and scan filtering are presented.

The NDT approximates the probability of having a reading at a certain position in a given cell j of a particular grid α by the Normal $N(\mu_{\alpha,j}, P_{\alpha,j})$. For clarity purposes, subindexes will be dropped throughout this section and the Normal distribution written as $N(\mu, P)$.

Both μ and P are computed using all the points lying inside the cell. Computing them in such a way may be problematic in the presence of outliers. If laser sensors are used, the number of outliers is so low that the algorithm has even to check whether the covariance matrix becomes near singular or not. However, if sonar sensors are used the number of outliers is not negligible. Moreover, the low angular resolution of these sensors tends to create dense regions of outliers, called *artifacts*. Artifacts are strongly dependant on the sonar beam incidence angle. In other words, artifacts may present different shapes depending on the robot pose. Thus, they should not be taken into account when computing the mean and the covariance of a given NDT cell because the NDT is assumed to be static with respect to x_B^A.

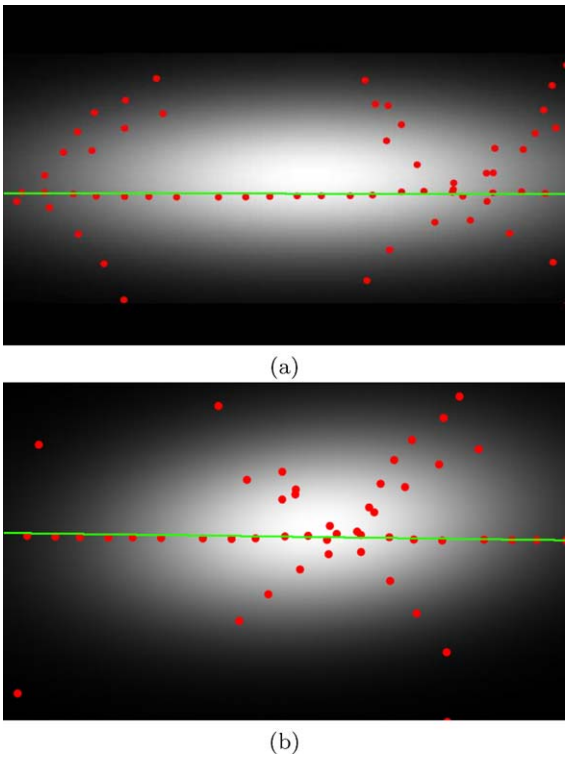

Fig. 5 The Normal distributions adjusting two sets of sonar readings containing (**a**) two artifacts and (**b**) one artifact. The *dots* represent the readings. The *line* represents the true location of the walls. Those readings not lying near the line correspond to artifacts. The grayscale images represent the Normal distributions

Figure 5 shows two sets of sonar readings and the Normal distributions adjusting them according to the NDT. Both sets of readings correspond to straight walls and both sets have artifacts. The influence of the artifacts is clear. In Fig. 5a, due to the two artifacts the mean has been moved upwards and the covariance along the Y axis is increased. In Fig. 5b, the artifact makes the Normal distribution to concentrate around it.

Our goal is, thus, to define a method to compute the mean and the covariance of the Normal distribution that better fits the inliers of a given set of sonar readings.

A common approach, especially in the computer vision community, to find the model that best fits to inliers (and discards outliers) of a given set of data points is the RANSAC.

This method has also been used in the robotics community to approximate sets of range readings by polygonal models.

RANSAC (*Random Sample Consensus*) was introduced by Fischler and Bolles (Fischler and Bolles 1981) as a new paradigm for fitting a model to experimental data containing a significant percentage of outliers.

The RANSAC algorithm is an iterative process. In each iteration, a subset of the original data points is randomly selected. These points are used to estimate the model that best fits them. Then the algorithm determines, for every point of the remaining set, how well the point fits to the estimated model. If the number of points that fit well to the estimated model is large enough, the algorithm ends and the model constitutes the output of the algorithm. Otherwise, the algorithm iterates.

The proposed RANSAC approach to fit a Normal distribution to a set of sonar readings will be referred to as RANSAC/GD (*RANSAC/Gaussian Determination*). It is described in Fig. 6. Line 11 of the algorithm requires further explanation. It is in charge of deciding whether the data point x fits the model defined by μ and P or not. Our proposal is to exploit the probabilistic meaning of the model to take this decision. Being the model $N(\mu, P)$ a bivariate Normal distribution, the following expression is a chi-squared distribution with two degrees of freedom:

$$D(x, \mu, P) = (x - \mu)^T P^{-1} (x - \mu). \tag{14}$$

To decide if the data point x fits the model, a confidence level β can be used. Thus, the point x is accepted if and only if $D(x, \mu, P) < \chi^2_{2,\beta}$. In our implementation, a confidence level $\beta = 0.99$ has been used, being $\chi^2_{2,0.99} \simeq 9.21$.

Line 20 also requires some explanation. In this line, a model is checked to decide whether it is better than any previous model or not. The decision is taken according to the eigenvalues of the covariance matrix. The underlying idea is that the sonar readings being analyzed are those contained in one sNDT cell. They are only a small part of a bigger set gathered in an environment where some continuity is supposed. Thus, it is reasonable to assume that in most cases the inliers of a given cell define a common structure with the inliers of one or more contiguous cells. Selecting the model according to the presented ratio between eigenvalues rewards those Normal distributions not concentrating around a specific value, but spreading along the dominant direction defined by the inliers.

Lines 29 to 33 also require further explanation. At this point, the estimated covariance is iteratively increased by δI (where I is the 2×2 identity matrix and δ a scalar so that $\delta < \lambda$) until the lowest eigenvalue is greater than λ times the biggest one. This adjustment has two main objectives. First, to prevent singular and near singular covariance matrices, as they could lead the minimization process to numerical problems. From this point of view, the value of λ should be, at least, 0.001 (Biber and Straßer 2003). The second goal is to prevent the Normal distribution PDF from

Algorithm:RANSAC/GD

Input:

Q:	Set of sonar data points
$nIter$:	Number of iterations to perform
n:	Number of points to be randomly selected
m:	Minimum number of points to consider a model
β:	Desired confidence level to accept a data point
λ:	Narrowness factor
δ:	Narrowness resolution

Output:

μ_{best}:	The estimated mean
P_{best}:	The estimated covariance
ε_{best}:	The quality of the estimated model
S_{best}:	The set of inliers used to build the model

```
1
2  begin
3      k ←0;
4      ε_best ←0;
5      while k < nIter do
6          S1 ← random selection of n data points from Q;
7          S2 ← ∅;
8          μ ←mean(S1);
9          P ←covariance(S1);
10         foreach x ∈ (Q − S1) do
11             if (x − μ)^T P^{-1} (x − μ) < χ²_{2,β} then
12                 S2 ← S2 ∪ {x};
13             end
14         end
15         if |S2| ≥ m then
16             μ' ←mean(S1 ∪ S2);
17             P' ←covariance(S1 ∪ S2);
18             ε_max ←max(eigenvalues(P'));
19             ε_min ←min(eigenvalues(P'));
20             if ε_max/ε_min > ε_best then
21                 ε_best ← ε_max/ε_min;
22                 μ_best ← μ';
23                 P_best ← p';
24                 S_best = S1 ∪ S2;
25             end
26         end
27     end
28     if ε_best > 0 then
29         repeat
30             ε_max ←max(eigenvalues(P_best));
31             ε_min ←min(eigenvalues(P_best));
32             P_best ← P_best + δI
33         until ε_min/ε_max < λ ;
34     end
35 end
```

Fig. 6 The RANSAC/GD algorithm

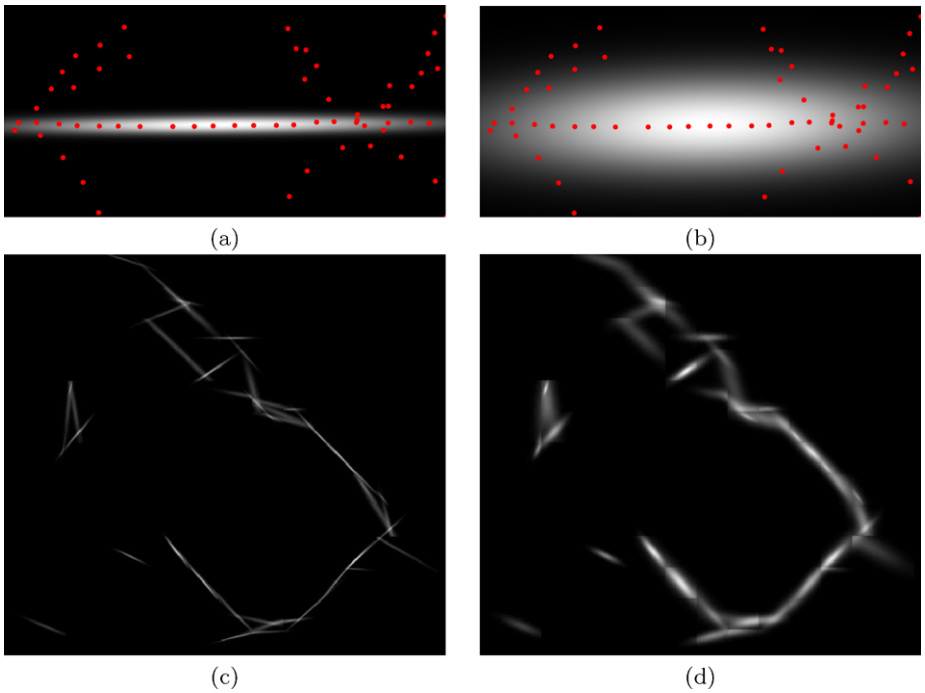

Fig. 7 The results of using RANSAC/GD. The effect of varying the narrowness factor λ can be observed. (**a**) Single cell with $\lambda = 0.001$. (**b**) Single cell with $\lambda = 0.05$. (**c**) Four overlapping grids with $\lambda = 0.001$. (**d**) Four overlapping grids with $\lambda = 0.05$

being too narrow in one direction. The problem of having very narrow PDFs appears when the initial estimate of the rototranslation between the scans is not very good. This is common when using sonar sensors, as the robot has moved a considerable distance to build the scans relying only on odometry. In those cases, a narrow PDF would lead to values close to zero in most of the addends of the score function (see Eq.13). Thus, it would be difficult for the minimization process to decide the direction towards which the minimum is located. In these cases, the value $\lambda = 0.001$ used with laser sensors would be problematic, and greater values are preferred. By experimentally tuning this parameter, we have observed that $\lambda \geq 0.1$ is required in the sonar case, and that the value of $\lambda = 0.5$ is the best choice for all the tested environments.

A final consideration has to be made regarding the selection of the parameters m, n and $nIter$. Although some details are provided in (Fischler and Bolles 1981) about how they should be selected, they strongly depend on the specific problem that is being solved. In our case, they have been chosen experimentally and are $n = 5$ (so that at least five points have to lie inside each NDT cell to be analyzed), $m = 0.35|Q|$ and $nIter = 1000$. Using these parameters, the obtained results are good and the algorithm runs fast.

Figure 7a and 7b show the Normal distributions that better fit the inliers of the same set of sonar readings shown in Fig. 5a. The effects of the narrowness factor λ can be appreciated: the bigger the narrowness factor, the bigger the dispersion along the smallest axis. The advantages of this approach are clear if one compares these images to the one obtained without using RANSAC/GD (Fig. 5a).

Figures 7c and 7d show the results of building the NDT grid using the RANSAC/GD approach with different values of λ. It can be observed that the use of RANSAC/GD produces an accurate representation of the environment and that the greater part of spurious readings have been discarded. The raw readings used to build the previous grids are shown in Fig. 8a. It can also be observed how the very narrow Likelihood Field in Fig. 7c may be problematic if a bad initial rototranslation is provided, as explained previously. In those cases, the narrow Likelihood Field would lead to values close to zero in most of the addends of the score function.

6.3 Filtering scans

As shown in the previous section, not all the readings belonging to S_{ref} have been taken into account when building the sNDT grid. Only the inliers selected by the RANSAC/GD have been taken into account. Thus, a similar approach should be applied to readings in S_{cur}. This approach will be referred to as RANSAC/GF (*RANSAC/Gaussian Filtering*).

RANSAC/GF is a technique to filter the readings in S_{cur} so that the generative process assumption is valid. As a result of this, S_{cur} will be divided in two subsets. The subset containing the readings that make the generative process assumption valid is referred to as the set of *accepted readings*. The subset containing the rest of the readings is referred to as the set of *rejected readings*.

The key idea behind the RANSAC/GF is similar to the sNDT grid building, but applied to S_{cur}. Given a grid's ori-

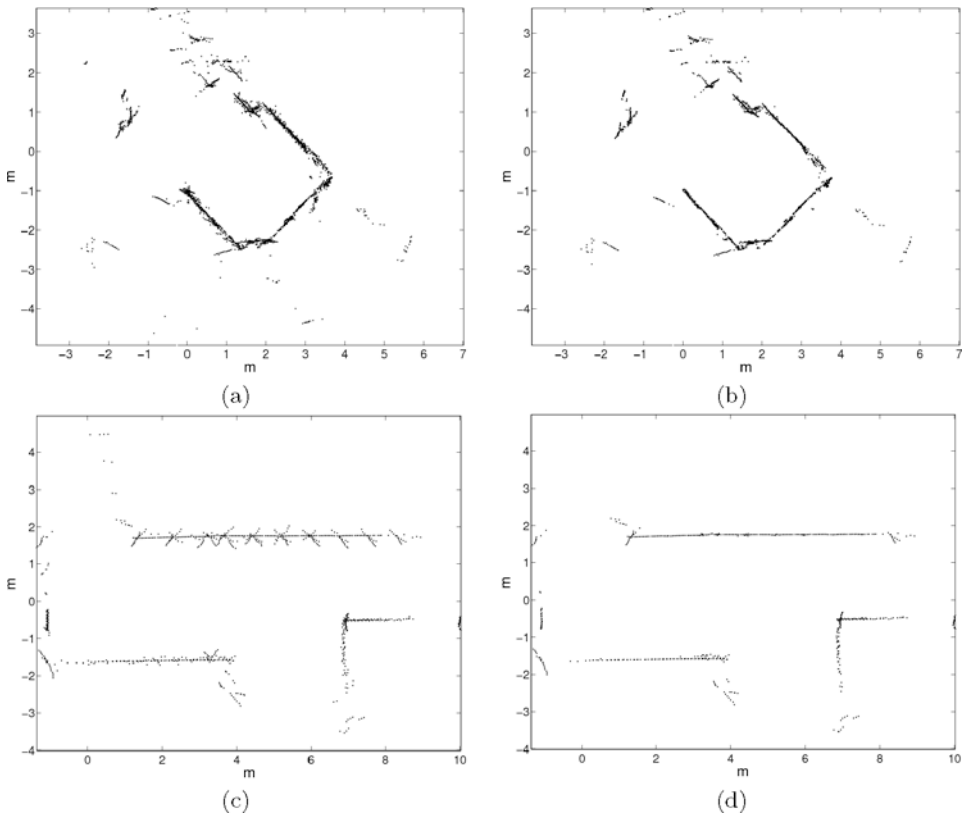

Fig. 8 RANSAC filtering. (**a**) and (**c**): original sets of sonar readings. (**b**) and (**d**): filtered sets of sonar readings

gin α, for each cell j let $\Omega'_{\alpha,j}$ denote the set of inliers selected by RANSAC/GD. That is, we define $\Omega'_{\alpha,j} \leftarrow S_{best}$ (see Fig. 6). Then, the set of accepted readings is built as follows:

$$\Omega' = \bigcup_{1\leq\alpha\leq 4} \bigcup_{1\leq j\leq N} \Omega'_{\alpha,j}. \tag{15}$$

In other words, the set of accepted readings is defined as the set of readings that have been selected as inliers by RANSAC/GD in any grid and cell when applied to S_{cur}. To ease notation, from now on, in the context of the sNDT, the term S_{cur} will refer to the set of accepted readings. In other words, $S_{cur} \leftarrow \Omega'$.

Figure 8 exemplifies the RANSAC/GF approach. It can be observed how isolated readings disappear after filtering. It can also be observed how spurious readings, especially those corresponding to artifacts, also disappear after filtering. The ability of RANSAC/GF to deal with artifacts can be explained as follows. Usually, artifacts are produced by small objects, such as door frames, located close to bigger environment structures, such as walls. When computing the Normal that better fits the readings in a cell containing an artifact, RANSAC/GF will most likely fit the Normal to the true obstacle rather than to the artifact. This is because the true obstacle produces more readings, which spread along one single dominant direction, than the artifact, which usually spreads in more than one direction.

This fact suggests two possible benefits from using this technique. On the one hand, to select only the reading belonging to S_{cur} that make the generative process assumption valid and allow the use of such readings in the sNDT while, on the other hand, to filter sonar readings so that clean sets could be used in other contexts, such as mapping or SLAM. Burguera et al. proposed in (Burguera et al. 2007a) a probabilistic approach to filter sonar data and demonstrated the benefits that could be provided in the context of ICP-based sonar scan matching (Burguera et al. 2007b). Barshan (Barshan 2007) proposed an approach to process ultrasonic arc maps and reviews other related techniques. Although some problems may appear in grid cells with a high amount of artifacts, the experimental results suggest that the RANSAC/GF approach could also be applied to process and filter sets of sonar readings in contexts similar to those of (Burguera et al. 2007a) and (Barshan 2007).

After building the sNDT grid using RANSAC/GD and filtering S_{cur} by means of RANSAC/GF, the matching process can be performed in the same way as in the original NDT (Biber and Straßer 2003).

7 The likelihood field/sum of Gaussians (LF/SoG)

The LF/SoG is an application of the Likelihood Field approach to scan matching. It approximates one of the scans by a sum of Gaussians. Thus, it builds a continuous and differentiable Likelihood Field. Although it involves more computations than the sNDT, a simplification is proposed to reduce the computational cost. The results provided by LF/SoG are better than those obtained with sNDT and NDT, as will be shown in Sect. 8.

7.1 Building the likelihood field

The Likelihood Field $f(p)$ is defined as follows:

$$f(p) = \sum_{q_j \in S_{ref}} \exp(-((p_x - q_{xj})^2 + (p_y - q_{yj})^2)) \qquad (16)$$

where $p = [p_x, p_y]^T$ and $q_j = [q_{xj}, q_{yj}]^T$. The exponent is minus the squared Euclidean distance between p and each point in S_{ref}. Thus, the function decreases with the distance to the points in S_{ref}. This fact is consistent with the Likelihood Field definition: the farther from an obstacle, the less likely one will have a reading. The reader should notice that each term in Eq. 16 corresponds to the not normalized PDF of a Normal distribution with covariance the identity matrix. Thus, the proposed Likelihood Field models the environment as a sum of Gaussians. That is why this Likelihood Field, as well as the Scan Matching technique that makes use of it, are called *LF/SoG*. Figure 9a shows an example of the Likelihood Field defined in Eq. 16.

Given the set of points p_i in S_{cur}, the correctness of a certain displacement and rotation $x = [x_x, x_y, \theta]^T$ between the two scans can be evaluated by the following score function:

$$s(x) = -\sum_{p_i \in S_{cur}} f(x \oplus p_i). \qquad (17)$$

The minus sign is used only to make further computations easier. The lower the value of $s(x)$, the better the transformation x is. Thus, the problem of scan matching using the Likelihood Field defined in Eq. 16 can be formulated as the problem of minimizing the score function shown in Eq. 17. It is important to point out that, in order to evaluate the score function $s(x)$, the Likelihood Field has to be computed only at m points, where m is the number of readings in S_{cur}.

7.2 Resampling

As the exponentials in Eq. 16 are summed, the value of the Likelihood Field is influenced by the density of the readings. The value of the Likelihood Field would be higher in the regions of S_{ref} with higher densities of readings. However, this is not a desirable situation, especially with sonar sensors. If sonar sensors have been used to gather the scan, the scan density strongly depends on parameters such as the robot speed or the material of the obstacle. To avoid this problem, it is necessary to resample the scan.

In this paper, we propose the replacement of small subsets of readings in S_{ref} by their center of gravity (Gutmann and Schlegel 1996). This approach is able to resample a scan with a minimal loss of information and has very low computational requirements. The idea behind this filter is to move a circular window over S_{ref} and substitute the readings inside the window with their center of gravity. Figures 9b and 9c exemplifies the effect of the resampling step. The radius of the window defines the minimum distance between the points in the resampled scan. This radius has to be defined experimentally. Low values for this parameter do not solve the influence of the readings density, while high values may render the resulting scan too sparse. By experimentally testing this parameter, we have observed that a value of 5 cm is a good choice.

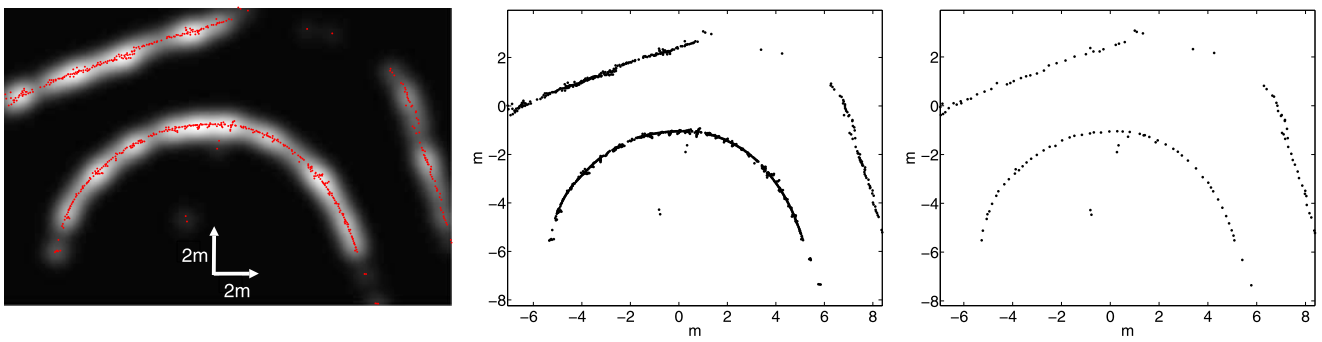

Fig. 9 (Color online) (**a**) Example of Likelihood Field (grayscale image) generated from a set of sonar readings (*red dots*). (**b**) Set of sonar readings before resampling. (**c**) Set of sonar readings after resampling. A window radius of 20 cm has been used to provide a clear representation

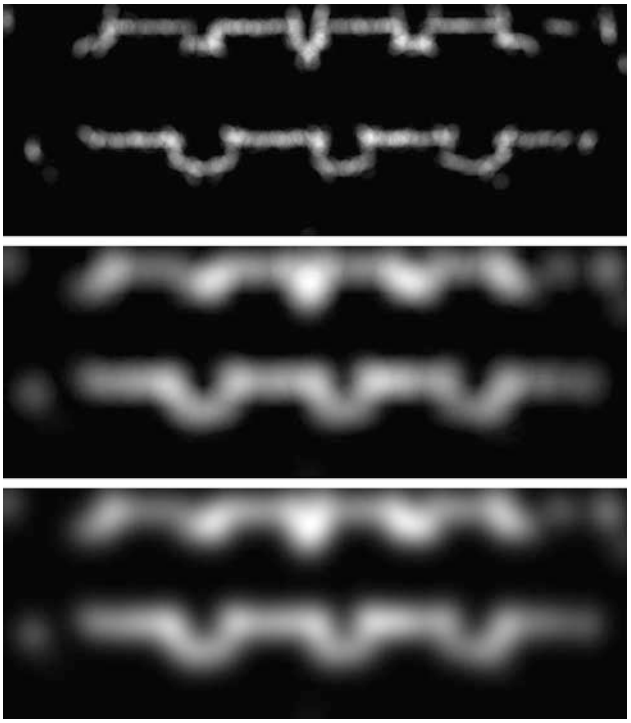

Fig. 10 Likelihood Field of a 12 m long fragment of a corridor using, from top to bottom, $\upsilon = 0.2$ m, $\upsilon = 0.6$ m and $\upsilon = 1$ m

7.3 Improving computations

One important advantage of the Likelihood Field approach with respect to ICP-based algorithms is that no correspondences are established. The whole set of points in S_{ref} is used. Thus, the contribution of each point $p_i \in S_{cur}$ to the overall matching process is influenced by the whole set of points in S_{ref}.

However, the value of $f(p)$ is barely influenced by those $q_i \in S_{ref}$ that are far from the point being evaluated. Thus, in order to reduce the computational cost, the addends of Eq. 16 involving points farther than a certain threshold υ from p could be neglected. Figure 10 shows the effect of different threshold values. In general, we have observed that the differences between the Likelihood Fields obtained with $\upsilon \geq 0.6$ m are not appreciable. This fact can be observed in Fig. 10 where there is not an appreciable difference between the Likelihood Fields corresponding to $\upsilon = 0.6$ m and $\upsilon = 1$ m. Consequently, from now on, the value of υ is set to 0.6 m.

7.4 Optimization

The estimated displacement and rotation is computed by minimizing Eq. 17. Our proposal is to minimize the mentioned score function by means of Newton's algorithm, described in Sect. 5.2.

The score function $s(x)$ is defined as a sum of exponentials. Thus, both the gradient vector and the Hessian matrix can be computed by summing the gradient vectors and the Hessian matrices of the addends respectively. Following explanations will focus on an addend involving p_i and q_j. To ease notation, in the context of this section we define $\alpha = [\alpha_1, \alpha_2]^T = [p'_{xi} - q_{xj}, p'_{yi} - q_{yj}]^T$ and $x = [x_x, x_y, \theta]^T = [t_1, t_2, t_3]^T$. Now, one addend of the score function $s(x)$ can be written as follows:

$$g(x) = -\exp(-\alpha^T \alpha) = -\exp(-(\alpha_1^2 + \alpha_2^2)). \tag{18}$$

Then, one addend of the gradient vector is as follows:

$$\nabla g(x_k) = \left.\frac{\partial g}{\partial x}\right|_{x_k, p_i, q_j} = 2\exp(-\alpha^T \alpha)\alpha^T J\big|_{x_k, p_i, q_j} \tag{19}$$

where J is the Jacobian matrix.

$$J = \left.\frac{\partial \alpha}{\partial x}\right|_{x_k, p_i} = \begin{bmatrix} 1 & 0 & -p_{xi}\sin\theta - p_{yi}\cos\theta \\ 0 & 1 & p_{xi}\cos\theta - p_{yi}\sin\theta \end{bmatrix}_{x_k, p_i}. \tag{20}$$

One addend of the Hessian matrix is a matrix of the form:

$$H = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix}_{x_k, p_i, q_j}. \tag{21}$$

Each term H_{rc} is as follows:

$$H_{rc} = 2\exp(-\alpha^T \alpha)\left(\left(-2\left(\alpha^T \frac{\partial \alpha}{\partial t_c}\right)\left(\alpha^T \frac{\partial \alpha}{\partial t_r}\right)\right) \times \left(\alpha^T \frac{\partial^2 \alpha}{\partial t_r t_c} + \left(\frac{\partial \alpha}{\partial t_r}\right)^T \left(\frac{\partial \alpha}{\partial t_c}\right)\right)\right) \tag{22}$$

where the terms $\frac{\partial \alpha}{\partial t_1}$, $\frac{\partial \alpha}{\partial t_2}$ and $\frac{\partial \alpha}{\partial t_3}$ correspond to the three column vectors in the Jacobian matrix of Eq. 20. The second partial derivatives of α are as follows:

$$\frac{\partial^2 \alpha}{\partial t_r t_c} = \begin{cases} \begin{bmatrix} -p_{xi}\cos\theta + p_{yi}\sin\theta \\ -p_{xi}\sin\theta - p_{yi}\cos\theta \end{bmatrix}, & r = c = 3, \\ \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \text{otherwise.} \end{cases} \tag{23}$$

The reader should notice that $g(x)$ (and, consequently, $f(p)$) are of class C^1 (i.e. continuously differentiable). Thus, the Hessian matrix is symmetric. Thanks to that, although the asymptotic complexity of the algorithm stays the same, the amount of computation required at each iteration can be slightly reduced.

There are some considerations regarding this minimization process. A requirement of Newton's method is that s has continuous first and second order partial derivatives in each region containing the x_k. As the score function s is a sum of exponentials, it meets the mentioned requirements. Thus, if all the points in S_{ref} are used to build the score function, $s(x)$ meets all the requirements of Newton's algorithm. However,

when using the method described in Sect. 7.3 to reduce the computational cost, the number of addends in $s(x)$ depend on the parameter x, rendering the score function non continuous. If Newton's method is applied in this situation, each iteration of the algorithm may be performed over different functions. This situation, where the function to be minimized changes, is similar to the minimization approach of ICP-based algorithms. However, if values of υ greater than 0.6 m are used as stated in Sect. 7.3, an addend is included or neglected from $s(x)$ only if its value is close to zero. Thus, the functions being minimized in two consecutive iterations would be *similar*, making possible in practice, the use of Newton's algorithm.

8 Experimental results

8.1 Overview

In this paper, two approaches to LF-based scan matching have been described: the sNDT and the LF/SoG. In order to evaluate these approaches, we compare them with other scan matching algorithms. On the one hand, they are compared to three ICP-based algorithms: sICP (*sonar ICP*), sIDC (*sonar IDC*) and spIC (*sonar probabilistic Iterative Correspondence*). On the other hand, they are also compared to the original NDT because it is a non ICP-based algorithm that has proved to be very effective when used with laser sensors.

Regarding the ICP-based algorithms, sICP and sIDC are the sonar versions of the well known and widely tested ICP and IDC algorithms. The only difference between the original versions and the sonar versions of ICP and IDC is that the Measurement Grouping and the Trajectory Correction steps, described in Sect. 4, are used.

The spIC is a probabilistic, ICP-based, sonar scan matching algorithm, which has proved to be more robust and accurate than sICP and sIDC. The main idea behind spIC is that correspondences are established if statistical compatibility exists between them. Also, spIC makes use of sonar models. To provide a fair comparison, exactly the same implementation of Measurement Grouping and Trajectory Correction has been applied to all the algorithms.

All the experiments discussed in this section have been carried out using real sonar data obtained with a Pioneer 3-DX mobile robot endowed with 16 Polaroid ultrasonic range finders. Data sets gathered in four different environments of our university have been used. The first environment has smooth stone walls, combined with glass walls and multiple door frames. The second environment has wooden walls. The NDT grid shown in Fig. 4 corresponds to this environment. The third environment is a corridor with rough walls and multiple entrances to offices. The Likelihood Field shown in Fig. 10 has been built using data gathered in this environment. Finally, the fourth environment is an unstructured room with cardboard boxes, chairs and tables. In consequence, the obtained data sets include structured and unstructured areas, posing different difficulties both to sonar sensing and odometry.

Two scans have been gathered along the same robot trajectory in each environment. Therefore, the displacement and rotation between both scans is perfectly known to be $[0, 0, 0]^T$. In other words, the *ground truth* is available. It is important to point out that although both scans have been gathered in similar conditions, they are not identical. Thus, they constitute a realistic test bench for sonar scan matching algorithms. Moreover, by gathering two scans at the same robot pose, the experiments concentrate specifically on the matching capabilities of the algorithms. In other words, by choosing a different rototranslation between two scans, the Measurement Grouping, the Trajectory Correction, and the errors involved in measuring the ground truth, would all influence the results.

Five experiments have been carried out introducing different initial location errors. Each experiment has been performed in each of the four described environments. The mentioned initial location errors correspond to the values assigned to the initial estimate, both in Newton's method and in ICP-based algorithms. These values have been randomly selected according to a uniform distribution between -0.05 m and 0.05 m in x and y, and between $-9°$ and $9°$ in θ in Experiment 1. The amount of initial error increases with the experiment, up to random errors between -0.25 m and 0.25 m in x and y and between $-45°$ and $45°$ in θ in Experiment 5. The procedure is repeated 1000 times per experiment and scan, which means a total of 20000 trials per algorithm. By analyzing the data obtained from these experiments, the algorithms are evaluated in terms of robustness, accuracy and convergence speed.

In all the experiments, the standard parametrization both for sNDT and NDT has been used ($L = 1$ m and $\lambda = 0.5$). The LF/SoG has been executed with $\upsilon = 0.6$.

8.2 Robustness

In order to evaluate the robustness of the algorithms, the results provided by each algorithm in each of the five experiments have been classified in four categories: *true positives*, *false positives*, *true negatives* and *false negatives*. A true positive appears when the algorithm converges to the right solution. A false positive describes those situations where the algorithm converges to a wrong solution. True negatives appear when the algorithm does not converge and the estimate generated in their last iteration was wrong. Finally, the situations where the algorithm does not converge, but where their last estimation was correct is described by false negatives. Although some ICP-based algorithms are convergent,

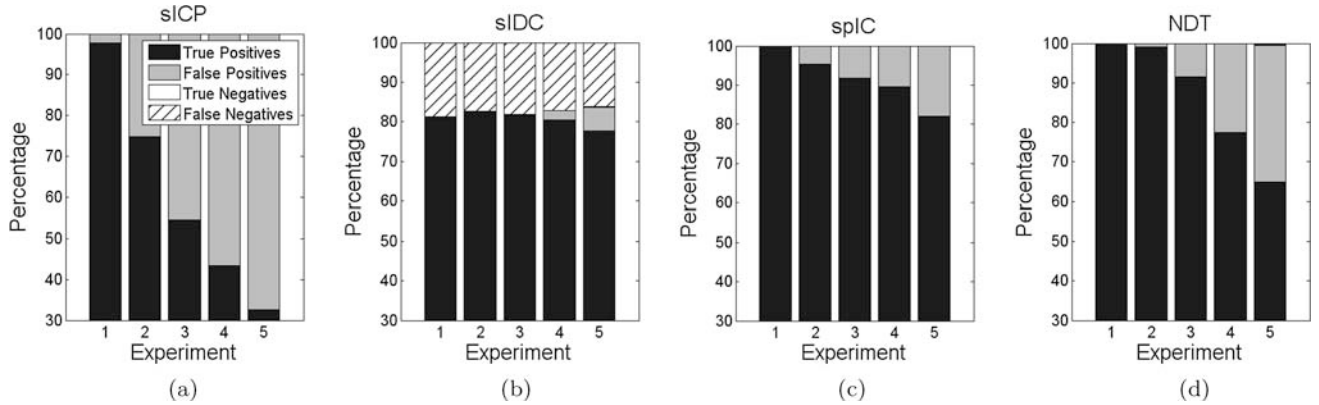

Fig. 11 Robustness of previously existing methods: (**a**) sICP, (**b**) sIDC, (**c**) spIC and (**d**) NDT. True positives (*black*), false positives (*gray*), true negatives (*white*) and false negatives (*lines*) are shown

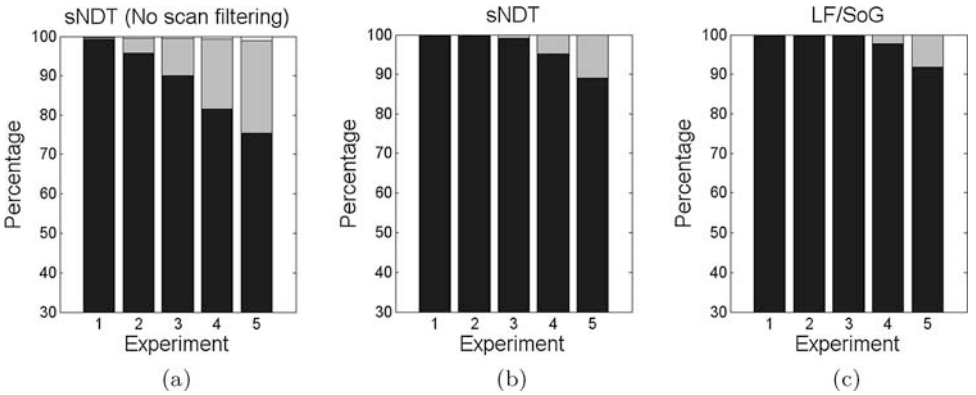

Fig. 12 Robustness of the methods presented in this paper: (**a**) sNDT without applying RANSAC/GF, (**b**) sNDT and (**c**) LF/SoG. True positives (*black*), false positives (*gray*), true negatives (*white*) and false negatives (*lines*) are shown

the term convergence, in the context of this section, refers to the ability of the algorithm to meet a numerical convergence criteria in a limited number of iterations.

To decide whether the solution provided by an algorithm is correct or not, the ground truth with a certain tolerance is used. We define as correct those results that are below 0.075 m in x and y, and below 0.075 rad in θ.

Figure 11 shows the robustness results for sICP, sIDC, spIC and NDT. These are previously existing methods, and the robustness results are only provided to compare them with those of the novel approaches discussed in this paper. It can be observed how sICP is a robust method in the case of very low initial errors. However, its robustness strongly decreases as the initial error increases. On the contrary, sIDC does not provide a number of true positives as high as sICP in presence of low initial error, but it tolerates bigger initial errors much better than sICP. The probabilistic approach, the spIC, is the best in terms of robustness, of all the tested ICP-based methods. Regarding the non ICP-based method (the NDT) it can be observed how it is even better in terms of robustness than spIC in the presence of low initial error. However, it is not able to cope with bigger errors and the robustness strongly decreases as the initial error increases.

Figure 12 shows the robustness results for the methods presented in this paper, the sNDT and the LF/SoG. The results regarding sNDT are divided in two blocks. First, those obtained when RANSAC/GD is applied to build the grid but no scan filtering is performed. Second, the results obtained using RANSAC/GD to build the grid and RANSAC/GF to filter S_{cur}. In this way, the effect of scan filtering can be observed.

It can be observed how, if scans are not filtered, the sNDT results are close to those of NDT. In presence of low and moderate initial errors, the NDT robustness is similar and even slightly better than the non filtered sNDT results. However, for larger initial errors, sNDT is shown to be significantly more robust than NDT. Thus, although in the presence of small initial errors the NDT seems to be slightly better than the non filtered sNDT, the non filtered sNDT provides a better global behavior in terms of robustness.

As described in Sect. 6.3, the use of RANSAC/GD classifies some of the readings in S_{ref} as outliers. These outliers are, consequently, not used in the grid building process, thus improving the grid's quality. However, by discarding readings in S_{ref} but not in S_{cur}, the generative process assumption is not valid. Then, RANSAC/GF was proposed. By applying RANSAC/GF, both S_{cur} and S_{ref} are treated un-

der identical conditions. The advantages of performing the mentioned filtering are clear when examining Fig. 12b. The filtered sNDT is remarkably more robust than the non filtered approach and the NDT. Moreover, the sNDT results are slightly better than those produced by spIC. For instance, in Experiment 5, the sNDT achieves a 89.07% of true positives whereas the percentage of spIC true positives is 82.07%.

A big advantage of the LF/SoG with respect to the sNDT, is that LF/SoG uses a continuous likelihood field. The effect of this fact on the robustness of the algorithm can be appreciated in Fig. 12c. It can be appreciated how LF/SoG is more robust than any other tested algorithm.

8.3 Accuracy

The accuracy is measured by means of the standard deviation of the errors of the positive estimates. By focusing on positive estimates, the experiment evaluates the accuracy in real situations where no ground truth is available and the algorithm only can distinguish between positives and negatives. A low standard deviation represents that each trial has produced results close to the ground truth, which means that good accuracy has been achieved. Higher standard deviations represent variations in the results obtained during the experiment, so that less accuracy has been achieved.

Figure 13 shows the standard deviations of the angular errors for different sonar scan matching algorithms. The angular error has been selected as it is commonly accepted that it constitutes a critical issue in localization. It can be observed in Fig. 13a how the most accurate of the previously existing algorithms is the spIC. Moreover, spIC has better accuracy than any of the algorithms presented in this paper.

Both the filtered and the non filtered approaches to sNDT provide higher accuracy than the original NDT. It is also clear that the use of RANSAC/GF to filter the scans in the sNDT strongly improves the accuracy. The LF/SoG algorithm is the best, in terms of accuracy, of the scan matching algorithms discussed in this paper. Moreover, LF/SoG accuracy is close to the spIC accuracy in presence of low and moderate initial error.

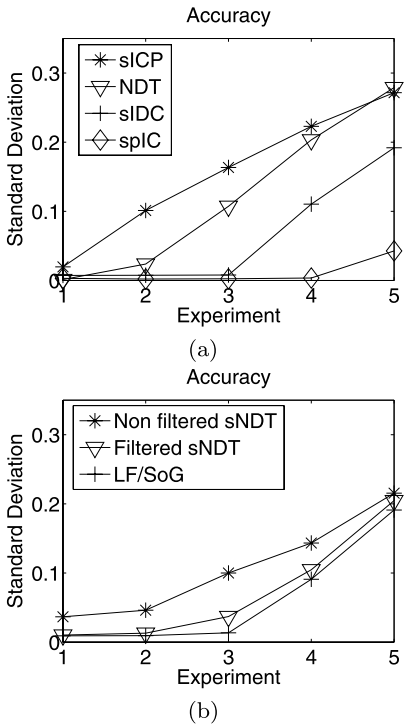

Fig. 13 Standard deviation of the errors. (**a**) Previously existing methods. (**b**) Methods presented in this paper

8.4 Convergence speed

The convergence speed is measured by means of the number of iterations required to achieve global convergence. Of course, the lower the number of iterations, the better the algorithm's performance.

The convergence speed results are shown for NDT, sNDT and LF/SoG, but not for ICP-based algorithms. An iteration in an LF-based algorithm corresponds to one step in Newton's algorithm. To the contrary, an iteration in an ICP-based algorithm corresponds to the minimization of one specific function. Thus, the number of iterations in an LF-based algorithm can not be compared to those of ICP-based algorithm. For this reason, only LF-based results are shown and analyzed here. Results regarding ICP-based algorithms are available in (Burguera et al. 2008).

Figure 14 shows the mean and the standard deviation of the true positive estimates for each of the five experiments. The graphical representation of the standard deviation has been reduced to 20% to provide a clear representation. Results show how even the non filtered sNDT achieves convergence faster than NDT in all the experiments when fed with sonar data. It can be observed how the use of RANSAC/GF to filter the scans makes the filtered sNDT converge faster than the non filtered approach. Finally, it is clear that the LF/SoG approach is able to achieve convergence much faster than any other of the presented methods.

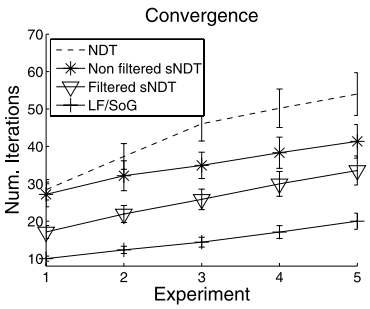

Fig. 14 Number of iterations to achieve convergence

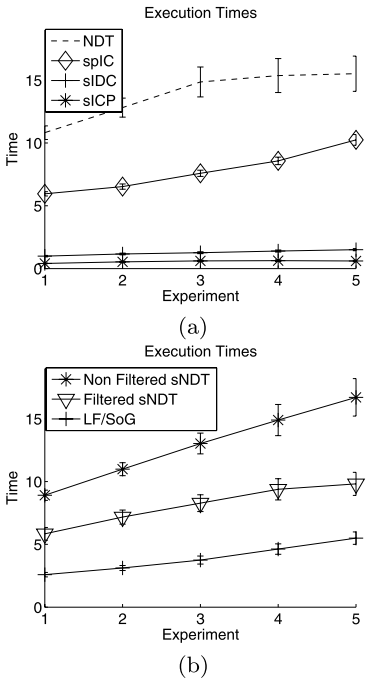

Fig. 15 Execution times (Matlab). (**a**) Previously existing methods. (**b**) Methods presented in this paper

In order to illustrate the previous convergence profiles and compare them to the ICP-based approaches, the execution times have also been measured. The mean and the standard deviation of the execution times of the true positive estimates for each of the five experiments have been computed. These results are shown in Fig. 15. The graphical representation of the standard deviation has been reduced to 20% in order to provide a clear representation. These execution times correspond to non optimized Matlab implementations. In addition, the results corresponding to NDT and sNDT do not include the time spent to build the grid. Thus, the interest of these results is not their absolute values but the relation between them, so that different algorithms can be compared.

Figure 15a shows the execution times for the previously existing algorithms. It can be observed how spIC and NDT, although being much more robust than sICP and sIDC also have higher time requirements. The results for the scan matching methods presented in this paper are shown in Fig. 15b. The Filtered sNDT is able to significantly reduce the computation time with respect to the NDT. This is due not only to the lower number of iterations required to achieve convergence, but also to the reduction of the number of readings in S_{cur} thanks to RANSAC/GF. Finally, it can be observed how the LF/SoG, which has shown to be the most robust of the tested algorithms and has a good accuracy also has very low computational requirements.

8.5 Visual examples

The presented experiments are aimed to provide a *quantitative* evaluation of the presented algorithms and to compare them to other existing, well known approaches to scan matching. One last experiment has been performed so that the algorithms can be evaluated *qualitatively*, by visual inspection. In this experiment, the robot moved inside a building of our university campus. Sonar scans were built by grouping the sonar readings along trajectories of 1.5 m. The results of performing localization using different algorithms are shown in Fig. 16. It can be observed how all the algorithms improve in some way the trajectory provided by odometry. However, sNDT and LF/SoG are the methods that provide better results. The LF/SoG is able to produce an almost straight corridor, thus, being closer to the real environment.

9 Conclusion

This paper presents a new approach to sonar scan matching. This approach requires a measurement grouping process, prior to the matching, to build dense sonar scans. Furthermore, after the matching process is completed, the whole trajectory involved in the grouping process is corrected as a result of a trajectory correction process.

The main contribution of the paper refers to the matching process. Our proposal is not to establish point to point correspondences. Thus, the algorithms discussed in this paper are not ICP-based. Instead, a function modeling the likelihood of having sonar readings at each spatial coordinate is built. This function is constructed according to the readings of one of the scans. As likelihood functions are being used, the new approach is called Likelihood Field Based scan matching (LF-based for short). After building the LF function, the readings in the other scan are matched against the LF function to find the displacement and rotation that better explains the robot motion.

Three different applications of the LF-based approach are presented. First, the non filtered sNDT: This application constitutes an improvement on the previously existing NDT to be used with sonar sensors. Second, the filtered sNDT: This application is similar to the non filtered sNDT, except that the sonar readings are filtered previous to the matching. Finally, the LF/SoG is proposed: This application defines a simple but effective Likelihood Field, having the advantage of being continuous and differentiable.

Experimental results evaluating the robustness, the accuracy and the convergence speed have been presented. The results demonstrate the validity of our approach by comparing it to well known ICP-based algorithms.

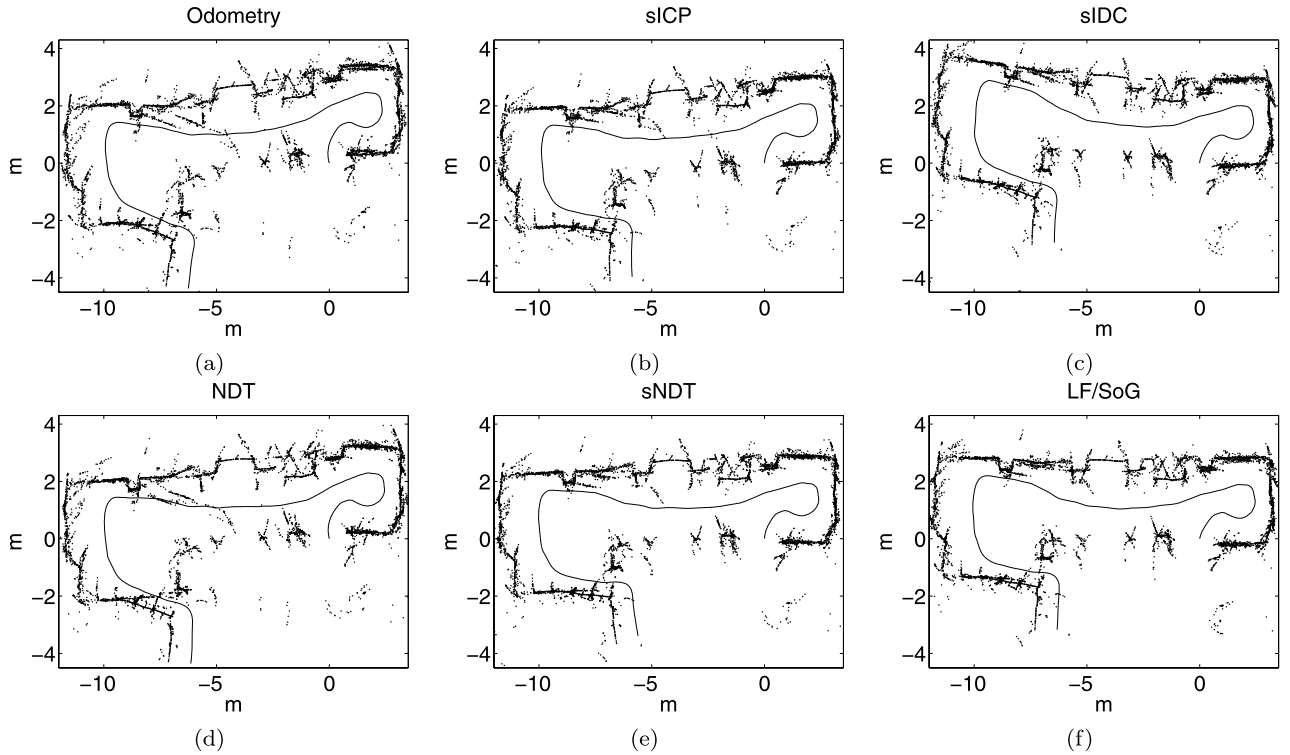

Fig. 16 Sonar readings for visual inspection. (**a**) Raw odometry, (**b**) sICP, (**c**) sIDC, (**d**) NDT, (**e**) sNDT, (**f**) LF/SoG

From our point of view, the LF-based approach constitutes a robust and accurate way to perform sonar scan matching. Moreover, this approach could easily be adapted to other range sensors. In particular, some noisy sensors, such as infrared range sensors, could benefit both from sNDT and LF/SoG, including the Measurement Grouping and the Trajectory Correction. Laser based scan matching could also profit from the LF/SoG. In that case, neither the Measurement Grouping nor the Trajectory Correction would be necessary. Also, as laser sensors are able to provide a complete scan from a single robot pose, the resampling step would not be necessary. Nevertheless, the LF/SoG Likelihood Field, as well as the whole formulation provided in the paper are of direct application to laser scan matching.

References

Barshan, B. (2007). Directional processing of ultrasonic arc maps and its comparison with existing techniques. *International Journal of Robotics Research*, *26*(8), 797–820.

Besl, P., & McKay, N. (1992). A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *14*(2), 239–256.

Biber, P. (2007). *Map building and localization for long-term operation of mobile robots in dynamic environments*. PhD thesis, Wilhelm-Schickard-Institut, University of Tuebingen, Germany.

Biber, P., & Straßer, W. (2003). The normal distribution transform: a new approach to laser scan matching. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)* (Vol. 3, pp. 2743–2748).

Biber, P., Fleck, S., & Straßer, W. (2004). A probabilistic framework for robust and accurate matching of point clouds. In *Proceedings of the 26th pattern recognition symposium (DAGM)*.

Bosse, M., Newman, P. M. Leonard, J.J. & Teller, S. (2004). SLAM in large-scale cyclic environments using the atlas framework. *International Journal of Robotics Research*, *23*(12), 1113–1139.

Burguera, A., Oliver, G., & Tardós, J. D. (2005). Robust scan matching localization using ultrasonic range finders. In *Proceedings of the conference on intelligent robots and systems (IROS)* (pp. 1451–1456), Edmonton, Canada.

Burguera, A., González, Y., & Oliver, G. (2007a), Probabilistic filtering of sonar data. In *IFAC symposium on intelligent autonomous vehicles (IAV)*, Toulouse, France.

Burguera, A., González, Y., & Oliver, G. (2007b). Probabilistic sonar filtering in scan matching localization. In: *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*. San Diego, USA.

Burguera, A., González, Y., & Oliver, G. (2007c). Probabilistic sonar scan matching for robust localization. In *Proceedings of the international conference on robotics and automation (ICRA)*, Rome, Italy.

Burguera, A., González, Y., & Oliver, G. (2008). A probabilistic framework for sonar scan matching localization. *Advanced Robotics*, *22*(11), 1223–1241.

Castellanos, J. A., Neira, J., & Tardós, J. D. (2001). Multisensor fusion for simultaneous localization and map building. *IEEE Transactions on Robotics and Automation*, *17*(6), 908–914.

Chen, Y., & Medioni, G. (1992). Object modelling by registration of multiple range images. *International Journal of Computer Vision and Image Understanding*, *10*(3), 145–155.

Dissanayake, G., Williams, S. B., Durrant-Whyte, H., & Bailey, T. (2002). Map management for efficient simultaneous localization and mapping (SLAM). *Autonomous Robots*, (*12*), 267–286.

Elfes, A. (1989). *Occupancy grids: a probabilistic framework for mobile robot perception and navigation*. PhD thesis, Dept. Electrical and Computer Engineering. Robotics Institute, Carnegie Mellon University.

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, *24*(6), 381–395.

Gutmann, J. S., & Schlegel, C. (1996). AMOS: comparison of scan matching approaches for self-localization in indoor environments. In *Proceedings of the Euromicro workshop in advanced mobile robotics*.

Hähnel, D., Burgard, W., & Thrun, S. (2003). Learning compact 3D models of indoor and outdoor environments with a mobile robot. *Robotics and Autonomous Systems*, *44*(1), 15–27.

Lee, D. (1996). *The map-building and exploration strategies of a simple sonar-equipped mobile robot*. Cambridge: Cambridge University Press.

Leonard, J., & Durrant-Whyte, H. F. (1992). *Directed sonar sensing for mobile robot navigation*. Cambridge: Kluwer Academic.

Lu, F., & Milios, E. (1997). Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent and Robotic Systems*, *18*(3), 249–275.

Minguez, J., Montesano, L., & Lamiraux, F. (2006). Metric-based iterative closest point scan matching for sensor displacement estimation. *IEEE Transactions on Robotics*, *22*(5), 1047–1054.

Montesano, L., Minguez, J., & Montano, L. (2005). Probabilistic scan matching for motion estimation in unstructured environments. In *Proceedings of the conference on intelligent robots and systems (IROS)* (pp. 1445–1450).

Moravec, H. P. (1988). Sensor fusion in certainty grids for mobile robots. *AI Magazine*, *9*, 61–74.

Pfister, S. T., Kreichbaum, K. L., Roumeliotis, S. I., & Burdick, J. W. (2002). Weighted range sensor matching algorithms for mobile robot displacement estimation. In *Proceedings of the IEEE international conference on robotics and automation (ICRA)* (Vol. 2, pp. 1667–1674).

Tardós, J. D., Neira, J., Newman, P. M., & Leonard, J. J. (2002). Robust mapping and localization in indoor environments using sonar data. *International Journal of Robotics Research*, *21*(4), 311–330.

Thrun, S. (2001). A probabilistic online mapping algorithm for teams of mobile robots. *International Journal of Robotics Research*, *20*(5), 335–363.

Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic robotics*. Cambridge: MIT Press.

Weiss, G., & Puttkamer, E. V. (1995). A map based on laserscans without geometric interpretation. In *Proceedings of intelligent autonomous systems (IAS)* (Vol. 4, pp. 403–407).

Zhang, Z. (1994). Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, *13*(2), 119–152.

Antoni Burguera received his degree in Computer Science from the Universitat de les Illes Balears in 2000. Since then, he has been Research Assistant of the Mathematics and Computer Science Department of the Universitat de les Illes Balears and has participated in projects mainly related with underwater robotics. He is currently a member of the Systems, Robotics and Vision Group, doing his PhD thesis on mobile robot localization and map building. His research interests include mobile robot navigation, localization and map building, focusing on the use of noisy sensors such as ultrasonic range finders.

Yolanda González received his MS degree in Computer Engineering from the Technical University of Deusto in 1992. From 1992 to 2000 he worked at the Open University of Spain (UNED) as a Research Assistant. She received the PhD degree in Electrical Engineering from this university in 1998. Since 2000 she is working as a Senior Researcher member of the SRV research group at the University of Balearic Islands. Her research interests include Scan Matching, SLAM, computer vision and robotics.

Gabriel Oliver received a degree in Physics from the Autonomous University of Barcelona. He earned his Ph.D. in Computer Science from the Polytechnic University of Catalonia. He is currently a senior lecturer at the Department of Mathematics and Computer Science at the University of the Balearic Islands where he is the Director of the Systems, Robotics and Computer Vision Group. His major research interests include computer vision and acoustic data processing for robot guidance and navigation. He has been the leading researcher of projects, related to underwater robotics, granted by the Spanish Scientific Council (CICYT).