Robust Planar Odometry based on Symmetric Range Flow and Multi-Scan Alignment

Mariano Jaimez, *Student Member, IEEE,* Javier Monroy, Manuel Lopez-Antequera, Javier Gonzalez-Jimenez, *Member, IEEE*

***Abstract*—This paper presents a dense method for estimating planar motion with a laser scanner. Starting from a symmetric representation of geometric consistency between scans, we derive a precise range flow constraint and express the motion of the scan observations as a function of the rigid motion of the scanner. In contrast to existing techniques, which align the incoming scan with either the previous one or the last selected keyscan, we propose a combined and efficient formulation to jointly align all these three scans at every iteration. This new formulation preserves the advantages of keyscan-based strategies but is more robust against suboptimal selection of keyscans and the presence of moving objects.**

An extensive evaluation of our method is presented with simulated and real data in both static and dynamic environments. Results show that our approach is one order of magnitude faster and significantly more accurate than existing methods in all the conducted experiments. With a runtime of about one millisecond, it is suitable for those robotic applications that require planar odometry with low computational cost. The code is available online as a ROS package.

***Index Terms*—Scan matching, range sensing, visual odometry.**

I. Introduction

MOTION estimation is one of the major challenges in robotics and computer vision. Virtually every robot, be it a drone, a humanoid or a manipulator, needs to accurately keep track of its position to perform an autonomous task. Although different technologies exist for estimating the motion of a robot (e.g. GPS systems, inertial sensors or encoders), visual odometry is arguably the most flexible and powerful solution since it can work with different input data (photometric/geometric) and can be adapted to almost any type of robot.

Among the many robotic platforms used nowadays, a significant percentage of them operate in structured environments and move on a planar surface. Examples of those are:

- Service robots working in hospitals, museums, hotels or airports [1].
- Telepresence robots that operate in domestic environments to monitor and assist old or disabled people [2], [3].
- Autonomous mobile robots employed in warehouses for sorting and delivery of goods [4].
- Modern vacuum cleaners like the iRobot Roomba or the Dyson 360 Eye.

To perceive their surroundings, these robots are often equipped with one or more laser scanners that allow them to survey the environment in a plane parallel to the floor. The data provided by these sensors is suitable for this kind of applications since it can be simultaneously exploited for obstacle avoidance, odometry, localization and 2D mapping.

In this paper we address the problem of estimating planar motion with a radial laser scanner. Our proposal takes inspiration from the latest research on dense and direct 3D visual odometry [5]–[8] and expresses the odometry as an energy minimization problem where the scans are aligned as piecewise continuous functions without searching for explicit correspondences. Despite the extensive body of literature in the field and the remarkable results achieved thus far, we demonstrate that this formulation provides more accurate results than existing techniques. Moreover, these results are achieved with a lower runtime (around 1 millisecond), which renders our method suitable for those robotic systems or applications that are computationally demanding and require real-time operation.

Our approach, which we will refer to as *Symmetric Range Flow-based Odometry* (SRF-Odometry), extends and improves the algorithm presented in [9]. That algorithm is based on the range flow constraint equation and formulates the motion of every observed point as a function of the velocity of the sensor, assuming that the environment is static. In this paper we build upon the same idea, and introduce the following contributions:

- A new symmetric formulation of geometric consistency between scans. To the best of our knowledge, this technique has been applied for the estimation of optical flow but never in the context of visual odometry.
- A new multi-scan formulation which combines the two standard techniques in visual odometry: alignment of consecutive scans/images and alignment against keyscan/keyframe.
- A procedure for modeling the accuracy of our algorithm as a function of the translation and rotation between the registered scans. Based on this model, we propose a new criterion for selecting keyscans by imposing thresholds on the maximum acceptable/desirable translational and rotational errors.

This work has been supported by the Spanish Government (project DPI2014-55826-R and grant program FPI-MICINN 2012), Andalusian Government (project TEP2012-530) and the ERC Consolidator Grant *3DReloaded*.

M. Jaimez is with the Department of Systems Engineering and Automation, University of Málaga, Spain, and with the Department of Computer Science, Technical University of Munich, Germany (email: marianojt@uma.es).

J. Monroy and J. Gonzalez-Jimenez are with the Department of Systems Engineering and Automation, University of Málaga, Spain (emails: jgmonroy, javiergonzalez@uma.es).

M. Lopez-Antequera is with the Department of Systems Engineering and Automation, University of Málaga, Spain, and with the Intelligent Systems group, University of Groningen, The Netherlands (email: mlopezantequera@uma.es).

- Faster and more accurate estimates than state-of-the-art techniques, both in static environments or in the presence of moving objects.

We present a thorough evaluation of our method with both synthetic and real data. We analyze how each of its main components contribute to its overall performance and test several versions of it (e.g. two-scan vs multi-scan alignment or robust optimization vs non-robust optimization). Furthermore, we compare our method with four state-of-the-art techniques [10]–[13]. Besides analyzing the results presented herein, we encourage the reader to watch the demonstration video and to test the algorithm by themselves. Both the video and the code, which is available as a ROS package, can be found at:

http://mapir.isa.uma.es/work/SRF-Odometry

II. Related Work

Over the last few decades, the scan matching problem has been extensively studied in robotics and computer vision. Although it can be regarded as a general problem, many of the proposed techniques focus on specific applications like localization [14] [15], SLAM [16] or odometry [17]. Since our interest is in the latter, this section will primarily consider those methods which have been particularly designed for (or are commonly applied to) the estimation of planar motion from a sequence of range scans.

In the context of 2D visual odometry, the majority of the existing approaches are based on a dense formulation, i.e., they use all the observations in the scans to align them. Sparse formulations based on interest points like FLIRT [18] or FALCO [19] have been employed for global pose optimization, localization and loop closure, but are rarely used for odometry.

Traditionally, ICP [20] or a number of its variants have been applied to solve the registration problem between consecutive scans. The *Iterative Dual Correspondence* method (IDC) [21] combines two different criteria to find correspondences between the scans: the standard closest-point rule and a new closest-range rule which leads to faster convergence thanks to a better estimation of rotations. Metric-based ICP (MB-ICP) [22] includes a new weighted angular term in its distance metric to improve the search for correspondences under rotation. In [22], MB-ICP obtains very accurate trajectory estimates when the robot wheel odometry provides the algorithm with an initial guess, but no information is provided about how these results would change if no external inputs (wheel odometry) were used. A different approach was proposed by Censi [10], where a point-to-line metric was used instead of the original point-to-point metric of ICP. Furthermore, the author presented an implementation which was an order of magnitude faster than existing ICP variants, while being more precise and efficient than the previous point-to-segment work in [23]. More recently, Generalized-ICP [24] improved the performance of existing ICP versions by including the covariance of both scans in the minimization problem (instead of using only that of the reference scan). However, Generalized-ICP has mainly been used for the registration of 3D point clouds and its performance in aligning 2D range scans does not seem to have been reported yet. In general, for this family of methods, accuracy depends on each particular version and implementation, yet they all share the same weakness: they tend to be computationally expensive.

Alternatively, other methods were specifically designed to solve the 2D scan matching problem:

- Gonzalez & Gutierrez [17] formulated the "velocity constraint equation", an adaptation of the optical flow constraint for range scans, and proposed estimating the lidar motion by imposing this restriction for every observation in the scans. However, their method was only tested on simple simulated scenarios and provided modest results.
- Remarkable results were presented by Biber & Strasser in [12]. Their method, named the *Normal Distributions Transform* (NDT), models the probability of finding a point at a certain position by using a collection of normal distributions to generate a piecewise continuous representation of the 2D plane. This model is created for the reference scan, and is used to evaluate the second scan by projecting it according to the estimated transformation. In this way, the NDT defines and minimizes a cost function which does not include the typical (and slow) search for correspondences. A similar idea based on the Distance Transform was presented by Fitzgibbon [25] to register 2D and 3D point sets and, more recently, a Signed Distance Function-based formulation was proposed by Fossel *et al.* [13] to solve the 2D SLAM problem for laser scanners.
- Censi *et al.* [26] proposed a new method based on the Hough Transform (HT) that permits the combining of the advantages of dense and feature-based scan matching algorithms. Their HT parameter space is the one associated with lines, and therefore the best results are achieved when the algorithm is tested in polygonal environments. Neither comparisons with other methods nor information about its runtime are provided.
- Diosi & Kleeman presented the Polar Scan Matching approach [27], where the translation and rotation between two scans are alternately estimated until convergence. In contrast to ICP, this method avoids the need to search for correspondences by simply matching points with the same bearing, resulting in better computational performance. This approach was subsequently extended and further evaluated in [11].
- The probabilistic method proposed by Olson [28] attempts to find the rigid transformation that maximizes the probability of obtaining the latest scan given the previous one. Additional information is used (control inputs or wheel odometry) to improve the method convergence and two different implementations, GPU and multi-resolution CPU, are presented. A thorough evaluation of its computational performance is included but, surprisingly, no results for the method's accuracy are presented.

More recently, other approaches have built upon the aforementioned works. This is the case for [29] and [30], which fuse laser odometry (Olson's laser odometry [28] and point-to-line ICP [10], respectively) with stereo vision to perform autonomous navigation with UAVs. Furthermore, the work of

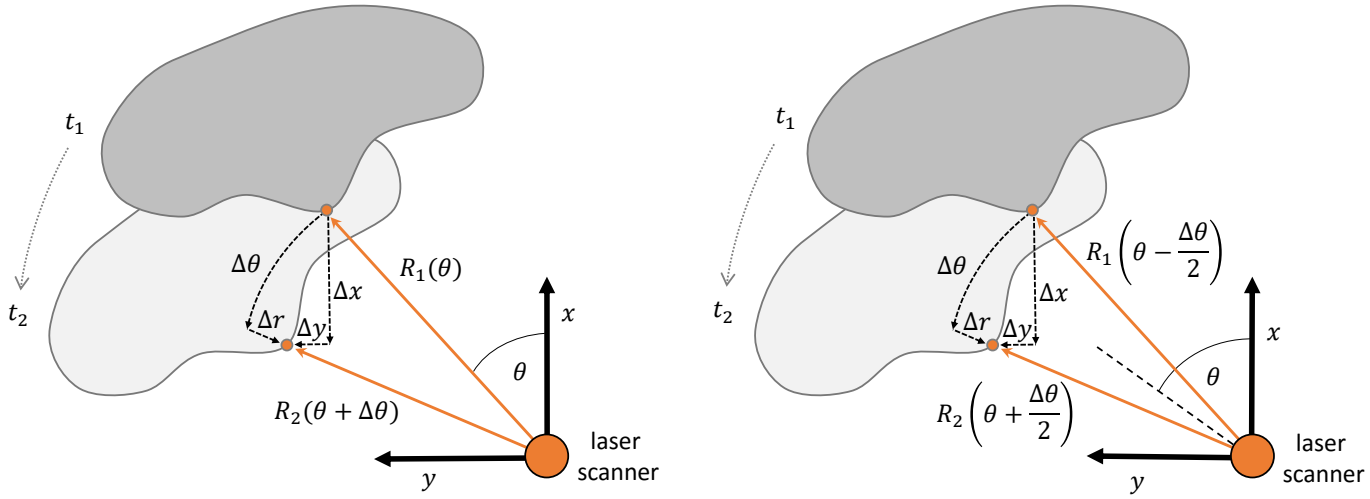

Fig. 1. Left: Standard formulation of geometric consistency between two scans R_1 and R_2 for a given point observed initially at θ. Right: Symmetric formulation of geometric consistency applied for the same point considered at the left scheme. The value of θ differs between the schemes because in the symmetric formulation θ represents the average of the initial and the final angles at which the point is observed.

Pomerleau *et al.* [31] presents a fast implementation and a thorough evaluation of some ICP variants using real-world 2D and 3D data sets.

III. Range Flow Constraint for Visual Odometry

In this section we derive a simple and linear constraint for the motion of the sensor by imposing geometric consistency between two consecutive scans. This constraint builds upon two main assumptions: the environment is static and the translation and rotation of the sensor are sufficiently small.

Let r, θ be the polar coordinates of a point with respect to the laser scanner and $R_1, R_2 : \Omega \to \mathbb{R}^+$ be two radial scans taken at consecutive instants of time t_1 and t_2, respectively. For simplicity, we assume that Ω is a continuous domain within the field of view (F_V) of the laser and is given directly in angular coordinates, i.e., $\Omega := [-F_V/2, F_V/2]$. During the time interval $[t_1, t_2]$, the apparent motion $\Delta r, \Delta\theta$ of any point of the environment with respect to the laser scanner must be consistent with the observations of the scans:

$$\Delta r = R_2(\theta + \Delta\theta) - R_1(\theta)\,. \tag{1}$$

This constraint is illustrated in Fig. 1 and is generally valid except in the case of occlusions. Often, this expression is linearized to obtain the so-called "range flow constraint" [9], [32]:

$$\Delta r = R_2(\theta) + \left.\frac{dR_2}{d\theta}\right|_\theta \Delta\theta - R_1(\theta) + O(\Delta\theta^2)\,, \tag{2}$$

$$O(\Delta\theta^2) = \frac{1}{2}\frac{d^2R_2}{d\theta^2}\Delta\theta^2 + O(\Delta\theta^3), \tag{3}$$

which is a geometric version of the well-known optical flow constraint if the second and higher order terms are neglected. This linearization (2) is also a particular 2D case of the general dynamic model presented in [33] for a laser rangefinder.

We propose using a slightly modified version of (2), where the motion is equidistributed between the two scans:

$$\Delta r = R_2(\theta + \Delta\theta/2) - R_1(\theta - \Delta\theta/2)\,. \tag{4}$$

This alternative representation (Fig. 1) has already been used in computer vision to estimate an inherently symmetric optical flow [34]. For us, the major advantage of this formulation is that its linearization is more precise than (2):

$$\Delta r = R_2(\theta) - R_1(\theta) + \left(\left.\frac{dR_2}{d\theta}\right|_\theta + \left.\frac{dR_1}{d\theta}\right|_\theta\right)\frac{\Delta\theta}{2} + O\left(\frac{\Delta\theta^2}{4}\right), \tag{5}$$

$$O\left(\frac{\Delta\theta^2}{4}\right) = \frac{\Delta\theta^2}{8}\left(\left.\frac{d^2R_2}{d\theta^2}\right|_\theta - \left.\frac{d^2R_1}{d\theta^2}\right|_\theta\right) + O\left(\frac{\Delta\theta^3}{8}\right). \tag{6}$$

As can be seen, this symmetric formulation requires more information than the standard range flow constraint (it requires the gradients of both scans) but it has a smaller linearization error for any given $\Delta\theta$.

Next we need to express the motion in Cartesian coordinates. The transformation from polar to Cartesian coordinates (x, y) is exact and linear if applied to instant velocities:

$$\dot r = \dot x\cos\theta + \dot y\sin\theta\,, \tag{7}$$

$$r\,\dot\theta = \dot y\cos\theta - \dot x\sin\theta\,. \tag{8}$$

To formulate (7) and (8) in terms of increments, they must be integrated between t_1 and t_2. By assuming that the displacements are small, we can approximate those integrals by the following linear terms:

$$\Delta r = \int_{t_1}^{t_2} \dot x\cos\theta + \dot y\sin\theta\; dt \approx \Delta x\cos\theta + \Delta y\sin\theta\,, \tag{9}$$

$$\Delta\theta = \int_{t_1}^{t_2} \frac{\dot y\cos\theta - \dot x\sin\theta}{r}\; dt \approx \frac{\Delta y\cos\theta - \Delta x\sin\theta}{\bar r}\,, \tag{10}$$

with $\bar r = (R_1(\theta) + R_2(\theta))/2$ being the best constant approximation of r between t_1 and t_2.

On the other hand, we need to impose the constraint that the relative motion between the environment and the sensor is only caused by the motion of the sensor itself (the environment

is static). This motion is encoded by the velocity vector $\boldsymbol{\xi}^s = (\xi^s_x, \xi^s_y, \xi^s_\omega)$, an element of the Lie algebra associated with 2D rigid transformations (i.e. $\boldsymbol{\xi}^s \in \mathfrak{se}(2)$). The motion of every point of the environment can thus be expressed as a function of $\boldsymbol{\xi}^s$ according to the kinematics of a rigid body:

$$\begin{pmatrix}\Delta x\\ \Delta y\end{pmatrix} = \int_{t_1}^{t_2} \begin{pmatrix}\dot{x}\\ \dot{y}\end{pmatrix} dt \approx \begin{pmatrix}-\xi^s_x + \bar{y}\,\xi^s_\omega\\ -\xi^s_y - \bar{x}\,\xi^s_\omega\end{pmatrix}, \tag{11}$$

where the average coordinates are computed as

$$\begin{pmatrix}\bar{x}\\ \bar{y}\end{pmatrix} = \frac{R_1(\theta) + R_2(\theta)}{2} \begin{pmatrix}\cos\theta\\ \sin\theta\end{pmatrix}. \tag{12}$$

Finally, plugging (9), (10) and (11) into the range flow (5) and discarding the higher order terms, we end up with a linear constraint for the motion of the sensor:

$$\left(\cos\theta + \frac{\bar{R}_\theta \sin\theta}{r}\right)\xi^s_x + \left(\sin\theta - \frac{\bar{R}_\theta\cos\theta}{\bar{r}}\right)\xi^s_y + \left(\bar{x}\sin\theta - \bar{y}\cos\theta - \bar{R}_\theta\right)\xi^s_\omega + R_2(\theta) - R_1(\theta) = 0\,, \tag{13}$$

where

$$\bar{R}_\theta = \frac{1}{2}\left(\left.\frac{dR_2}{d\theta}\right|_\theta + \left.\frac{dR_1}{d\theta}\right|_\theta\right) \tag{14}$$

is the average derivative of the two consecutive scans. Therefore, the motion of the sensor $\boldsymbol{\xi}^s$ can be obtained by matching consecutive scans (which should be differentiable or piecewise differentiable) without searching for and aligning explicit correspondences.

As previously stated, this derivation is valid under the assumption of small motions, i.e. those for which the linearization (5) holds. Although there is no sharp transition between "small" and "large" motions, we generally consider that the motion is small if $\Delta\theta$ is always less than or equal to the local neighbourhood used to approximate the range gradients $\bar{R}_\theta$. Commonly, these gradients are approximated with a centred formula using the values of the following and previous observations and, therefore, the angular increment $\Delta\theta$ should be less than or equal to the angle between contiguous observations in the scan.

IV. Optimization Problem

Theoretically, three independent constraints would suffice to obtain the lidar motion but in practice this is unfeasible because (13) tends to be inexact due to the noise of the measurements, the errors made by the linear approximation (5) or the presence of moving objects (non-static environment). Therefore, we use a dense formulation in which all the scan observations contribute to the motion estimate.

The geometric residual $\rho(\boldsymbol{\xi}, \theta)$ is defined as the evaluation of the range flow constraint (13) for a given motion $\boldsymbol{\xi}$ at a given angle θ:

$$\rho(\boldsymbol{\xi},\theta) = R_2(\theta) - R_1(\theta) + \left(\bar{x}\sin\theta - \bar{y}\cos\theta - \bar{R}_\theta\right)\xi_\omega + \left(\cos\theta + \frac{\bar{R}_\theta\sin\theta}{\bar{r}}\right)\xi_x + \left(\sin\theta - \frac{\bar{R}_\theta\cos\theta}{\bar{r}}\right)\xi_y. \tag{15}$$

Since not every arbitrary angle θ can be evaluated, rather only those sampled by the laser scanner, we simplify notation and use $\rho_n(\boldsymbol{\xi})$ to refer to the residual associated with the n-th observation of the scans. To obtain an accurate motion estimate, all the geometric residuals are minimized within a robust cost function:

$$\boldsymbol{\xi}^M = \arg\min_{\boldsymbol{\xi}} \sum_{n=1}^{N} F\big(\rho_n(\boldsymbol{\xi})\big)\,, \tag{16}$$

$$F(\rho) = \begin{cases} \frac{\rho^2}{2}\left(1 - \frac{\rho^2}{2c^2}\right) & |\rho| \le c \\ \frac{c^2}{4} & |\rho| > c \end{cases}, \tag{17}$$

N being the number of points in the scan. The function $F(\rho)$ is a smooth version of a truncated parabola (Fig. 2), and c is an adjustable parameter. $F(\rho)$ is continuous and differentiable everywhere, and becomes flat for residuals higher than c, which represents an effective and automatic way to downweight (or even discard) outliers. The parameter c is computed as a ratio of the median absolute deviation (MAD) of the residuals (see Section VIII-C).

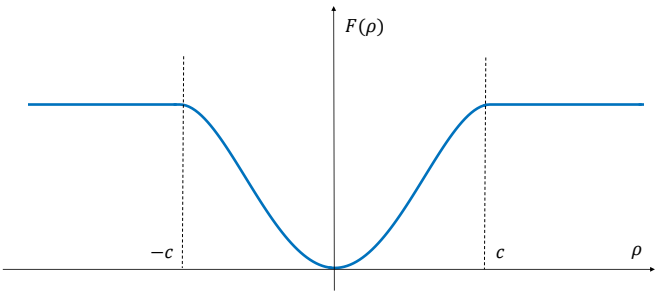

Fig. 2. Robust penalty function employed to minimize the geometric residuals.

A. Pre-Weighting Strategy

There are some factors that can render (13) inaccurate, mainly the unfulfillment of the rigidity hypothesis (11) and deviations from the linear approximation made in (5). Although the robust function $F(\rho)$ can alleviate their effect on the overall motion estimate, it does not eliminate it completely. The presence of moving objects is hard to detect before solving the system and we therefore rely on the robust function $F(\rho)$ to downweight them during the minimization process. On the other hand, deviations from the linear approximation adopted in (5) can be detected beforehand, which helps to accelerate convergence in (16) and also leads to more accurate results. For this purpose, we propose a pre-weighting strategy to downweight the residuals of those observations where the range function (4) is highly nonlinear or even non-differentiable. We call it "pre-weighting" because it is applied before the minimization problem (16) is solved.

In order to quantify the error associated with the linearization of (5), we evaluate the second order terms (6) of the Taylor series. Moreover, it is important to identify those regions of the scans where the range function is not only nonlinear but also non-differentiable. These regions are mainly the edges of the various observed objects, and are typically characterized by very high values of the first order derivatives, both the angular $\bar{R}_\theta$ and the temporal $R_t = R_2 - R_1$. To penalize

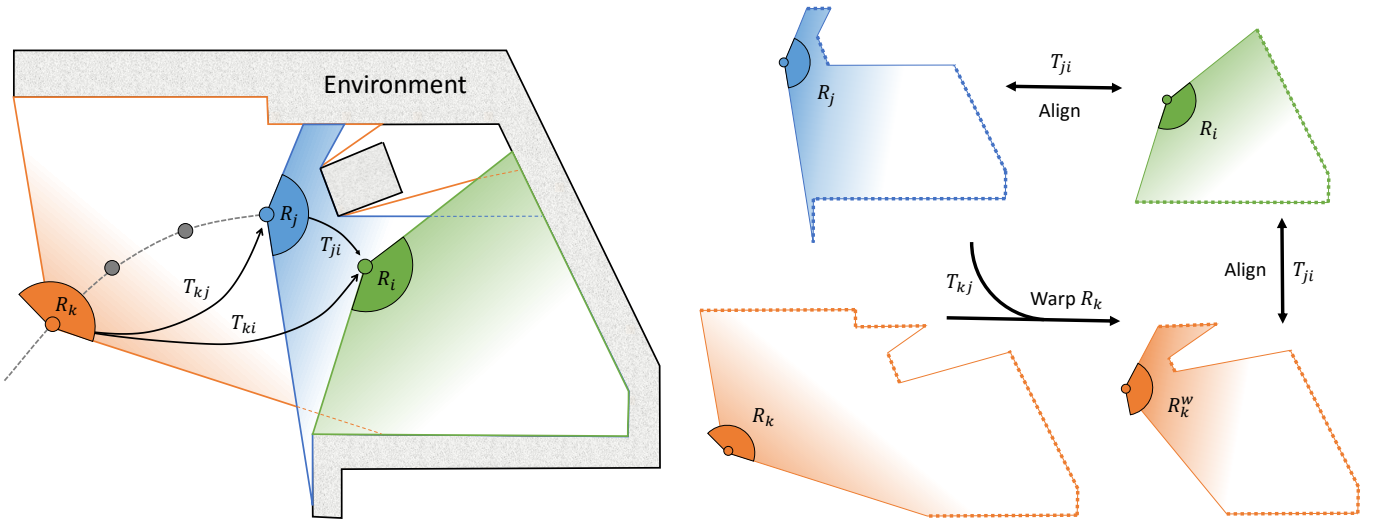

Fig. 3. Left: Schematic representation of our three-scan formulation, with the last scan R_i shown in green, the previous one R_j in blue and the keyscan R_k in orange. Right: Diagram of the proposed optimization problem. First, the keyscan R_k needs to be warped according to T_{kj} and, afterwards, both R_k^w and R_j are aligned with R_i. Notice that the warping for R_k^w was performed by keeping the most distant points of R_k after projection, which are likely to represent the structure of the environment and will also provide additional information when compared to R_j.

these two effects, nonlinearities and discontinuities, we define the following pre-weighting function:

$$w = \frac{1}{\sigma_s^2 + K_D\left(\bar{R}_\theta^2 + R_t^2\right) + K_{2D}\bar{R}_{\theta\theta}^2}\,, \tag{18}$$

where $\bar{R}_{\theta\theta}$ is the averaged second-order derivative of R_1 and R_2. The parameters K_D, K_{2D} quantify the relative importance of first and second order derivatives. Furthermore, we add an additional term σ_s^2 to model the noise of the measurements. In this paper we employ a simple constant value for σ_s, but more elaborate and precise noise models could be used instead.

In summary, to estimate the sensor motion we initially compute a pre-weighted set of residuals

$$\rho_n^w(\boldsymbol{\xi}) = w_n\ \rho_n(\boldsymbol{\xi}) \qquad\qquad n \in \{1, 2...N\} \tag{19}$$

which are subsequently minimized according to (16),(17).

V. Multi-Scan Formulation

Pure odometry always estimates motion between consecutive sets of input data, irrespective of whether these data are wheel rotations, RGB images or range scans. However, this purely incremental strategy has one major drawback in visual odometry: every new increment introduces some error in the pose estimate, even if the real motion is very small or null. This deficiency is commonly solved by periodically selecting a particular scan of the sequence, named as "reference scan" or "keyscan", and aligning every new scan against it. This keyscan acts as a local anchor, helping to reduce the drift of the estimated trajectory. When the incoming scans get too far from the selected keyscan, in the sense that there is not much overlap between the two, a new keyscan must be set and the process continues. A keyscan-based formulation is typically more accurate than purely incremental estimation, but strongly depends on the criterion used to introduce new keyscans. This criterion must prevent the inserting of redundant keyscans and, above all, it must guarantee that the latest scan is always close enough to the keyscan so that they can be aligned.

In this paper we propose a hybrid formulation in which the latest scan is aligned simultaneously against the previous scan and against a keyscan (Fig. 3). This strategy preserves the advantages of a keyscan-based approach while at the same time reducing the risks originating from inappropriate selection of keyscans. Since three different scans are now fed to the algorithm, the detrimental effects of the sensor noise and the presence of moving objects are also alleviated.

Let $T_{ki}, T_{kj} \in$ SE(2) be the homogeneous transformations between the scans R_i and R_j $(j = i-1)$ and the last keyscan R_k respectively, and let $T_{ji} \in$ SE(2) be the incremental transformation between R_i and R_j (see Fig. 3). Since these transformations form a loop, the following constraint must be fulfilled:

$$T_{kj}\, T_{ji} = T_{ki}\,, \tag{20}$$

where T_{kj} is assumed to be known from the previous estimation. Therefore, we can build an optimization problem for jointly aligning the scans R_k, R_j with R_i, subject to (20). There are two possible ways of formulating this problem:

- Optimizing for the two sets of unknowns explicitly , i.e., for the vectors $\boldsymbol{\xi}_{ji}$ and $\boldsymbol{\xi}_{ki}$ associated with T_{ji} and T_{ki}.
- Warping R_k towards R_j according to the already known transformation T_{kj} and solving for T_{ji} in both cases.

The second option is more efficient because it involves less unknowns (we only need to estimate $\boldsymbol{\xi}_{ij}$) and implicitly imposes the constraint (20) through the warping of R_k towards R_j. Thus, the optimization problem associated with the proposed multi-scan formulation is:

$$\boldsymbol{\xi}^M = \arg\min_{\boldsymbol{\xi}} \sum_{n=1}^{N} F(w_n^{ji}\rho_n^{ji}(\boldsymbol{\xi})) + F(w_n^{ki}\rho_n^{ki}(\boldsymbol{\xi}))\ . \tag{21}$$

VI. SOLVER

Our motion estimation problem is nonlinear and non-convex because both the original constraint of geometric consistency (4) and the robust function $F(\rho)$ (17) are nonlinear and non-convex. Where the constraint of geometric consistency is concerned, this limitation is solved by deriving the range flow equation (5) and defining the geometric residuals as linear functions of the motion of the lidar. However, the resulting linear constraints are only valid for very small displacements and/or rotations and would fail to estimate real motions in practice. However, this issue has already been addressed in the literature and can be solved by formulating the motion estimation problem within a coarse-to-fine scheme. In a coarse-to-fine scheme, two pyramids of scans (or typically images) are built and aligned, starting from the coarsest level, where the linearization (5) holds for larger displacements, and then following this by subsequent refinements in levels with increasing resolutions. Thus, each level incrementally improves the alignment and leaves "less motion" to estimate in the remaining (and finer) levels. More details about coarse-to-fine strategies and warping can be found in [9], [35].

At each level of the coarse-to-fine scheme, the optimization problem (21) is solved using Iteratively Reweighted Least Squares (IRLS), where the weights associated with the smooth truncated parabola $F(\rho)$ are:

$$W(\rho) = \begin{cases} 1 - \frac{\rho^2}{c^2} & |\rho| \leq c \\ 0 & |\rho| > c \end{cases}, \tag{22}$$

Since the robust function $F(\rho)$ is non-convex, we have also contemplated and tested other alternatives for optimizing (21), like the lifting strategy proposed in [36], but they do not improve results if compared to IRLS and involve more complicated and slower implementations.

Lastly, for the coarser levels of the coarse-to-fine scheme, it can occur that the motion to be estimated is actually outside the range of motions for which the linearization (5) holds. In this case, two different outcomes are possible:

- The real motion is much larger than the valid range for the linearization, and therefore the solver will provide a completely wrong solution.
- The real motion is out of but close to the valid range for the linearization, and therefore the solver will provide a solution which is not precise but comes close to the real motion.

The big failure of the first case cannot be avoided: the scans are simply too far apart and the algorithm will fail. Nevertheless, in the second case the solution can be used to warp one of the two scans towards the other, creating a new configuration in which the remaining motion (to be estimated) is smaller and, hence, can be obtained more precisely by re-running the algorithm. This process of estimating motion and warping scans can be performed iteratively as long as the last estimated motion is larger than a given threshold ϵ, improving the basin of convergence of our method. The whole estimation process is summarized in Algorithm 1, where the operator $\oplus$ represents

$$\boldsymbol{\xi}_a \oplus \boldsymbol{\xi}_b = \log\left(\exp\left(\hat{\xi}_a\right) \cdot \exp\left(\hat{\xi}_b\right)\right) \tag{23}$$

and the $\hat{\xi}$ is the skew-symmetric matrix associated to $\boldsymbol{\xi}$.

Algorithm 1 Motion estimation in a coarse-to-fine scheme

```
Build Scan Pyramids
Initialize estimated motion: ξ^S = 0
for l = 1 : number of levels do
    Initialize motion in this level: ξ_l^S = 0
    for m = 1 : max iterations do
        Compute ξ_l^M solving (21) with IRLS
        Update ξ_l^S = ξ_l^S ⊕ ξ_l^M
        Warp R_i → R_j according to ξ_l^S
        if ||ξ_l^M|| < ε then break
    Update ξ^S = ξ^S ⊕ ξ_l^S
    Warp R_i → R_j according to ξ^S
```

VII. KEYSCAN SELECTION

As previously mentioned, a keyscan-based formulation provides more accurate trajectory estimates than consecutive scan alignment, but requires a suitable keyscan selection criterion. Different strategies have been proposed to introduce new keyscans (or keyframes) when a certain magnitude exceeds a manually set threshold. Typically, this threshold is applied to the estimated angular and linear displacement [37], the residual after image alignment [38] or the entropy of the estimation [39]. The main problem with these approaches is that the selection of thresholds is not directly related to the final performance of the algorithm and, hence, the pose estimation error is not constrained to a well defined range. Consequently, these approaches involve tedious trial-and-error stages to tune their thresholds until the desired performance is achieved.

In this work we propose to model the pose estimation error of our algorithm as a function of the translation and rotation between the registered scans. Using this model, thresholds for the keyscan selection can be set directly over the error domain such that a maximum rotation and/or translation error is not surpassed. These thresholds define a 2D working region over the translation and rotation domains, which will be used during operation to trigger the selection of new keyscans.[1]

When modeling the error, the specifications of the laser scanner (field of view, maximum range and number of points in the scan) are the main source of performance variability. This means that it is not possible to obtain a single and universal error model, but instead the model must be tuned according to the characteristics of the scanner employed. In this section we obtain the error models for two different laser rangefinders: a Hokuyo UTM-30LX and a SICK LMS-500 (see Table I), used in the simulated and real experiments presented in Section IX.

[1]Notice that the main virtue of this approach resides in the easy and efficient selection of keyscans based on the desired tracking performance (expressed as maximum allowed translational and rotational drift). Yet, as for any other keyscan selection criterion, the estimate error cannot be proven to be bounded to a particular figure and ultimately depends on the quality of the data used to obtain the imposed thresholds, the environment and other factors such as sensor tilting.

TABLE I
RANGEFINDERS USED FOR MODELING THE ERROR OF THE ALGORITHM.

Model	Samples	FOV (deg)	Range (m)
Hokuyo UTM-30LX	1080	270	30
SICK LMS-500	361	180	80

We rely on simulated experiments with precise ground truth to generate sufficient and varied samples to model the estimation error. The data is generated by simulating a laser scanner in a certain environment and applying random displacements and rotations to it in the range of $[0, 1]$ metres and $[0, 1]$ radians. For these simulations we make use of three publicly available scenarios/maps: Belgioioso Castle and Intel Research Lab from the Robotics Data Set Repository (Radish) [40], and the Sarmis domestic environment from Robot@home dataset [41]. Furthermore, Gaussian noise with σ ranging from 5 mm to 25 mm is added to the laser measurements. Overall, 60,000 odometry estimates (i.e 20,000 error samples per scenario) are employed to model the error.

After collecting the samples, we estimate the translational e^T and rotational e^R errors at any location (x^T, x^R) of the translation-rotation plane by calculating a weighted mean of the errors obtained in simulation using an anisotropic Gaussian windowing function with $\sigma_R = 0.1$ rad and $\sigma_T = 0.12$ m:

$$e^T(x^T, x^R) = \frac{\sum_{s=1}^{S} \lambda_s e_s^T}{\sum_{s=1}^{S} \lambda_s}, \tag{24}$$

$$e^R(x^T, x^R) = \frac{\sum_{s=1}^{S} \lambda_s e_s^R}{\sum_{s=1}^{S} \lambda_s}, \tag{25}$$

$$\lambda_s = \exp\left(-\frac{||x^T - x_s^T||^2}{2\sigma_T^2} - \frac{||x^R - x_s^R||^2}{2\sigma_R^2}\right) \tag{26}$$

where S is the total number of samples. The resulting surfaces are shown in Fig. 4. As expected, when the translation and/or rotation between consecutive scans increases, so does the average error in the pose estimation. Comparing the error models of the two laser scanners, we can see that they both have a similar shape, but the error values are higher for the SICK LMS-500. This is to be expected because the SICK rangefinder has a smaller field of view and fewer points per scan than the Hokuyo scanner.

Making use of these models, we set thresholds directly for the translational and rotational errors to restrict them to an acceptable range. Specifically, in this work we set these thresholds to 10 mm and 0.1 degrees, respectively. By intersecting them with the surfaces of the error models, we obtain the final working regions shown in Fig. 5. It can be seen that the shapes of the regions are similar but their scales differ. The scanner with larger field of view and higher number of points offers a larger working region, indicating that scans carrying more information can work with sparser keyscans while keeping the pose error within the same bounds. Once the working region has been calculated, we fit a fourth-degree polynomial to its boundary, which will be evaluated at each iteration to determine whether the system is inside or outside of the working region. If the estimated pose with respect to the current keyscan falls within the working region, we trust the odometry estimation and keep the current keyscan while otherwise we trigger a keyscan update.

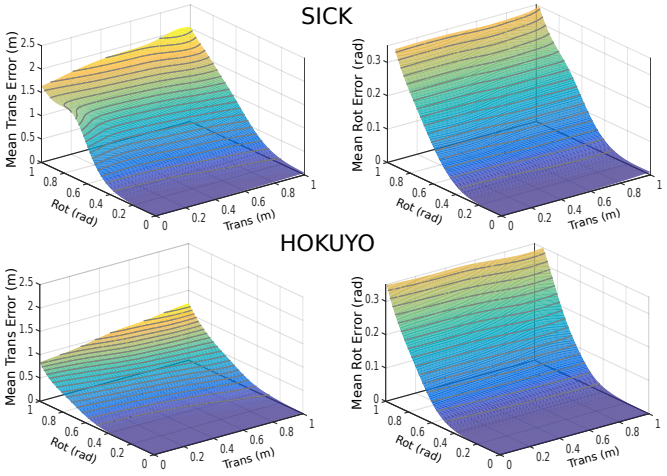

Fig. 4. Translation and rotation error models of the presented odometry algorithm when employing: (top) SICK LMS-500, and (bottom) Hokuyo UTM-30LX.

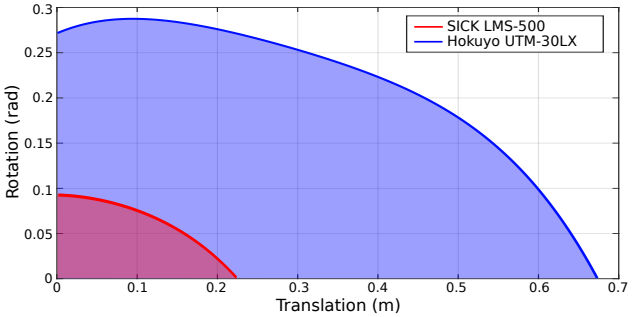

Fig. 5. Working regions obtained from the error models (see Fig. 4) for two different laser rangefinders, setting a maximum translation error of 10 mm, and a maximum rotation error of 0.1 degrees. If an odometry estimate falls outside the working region, a keyscan update is triggered.

VIII. IMPLEMENTATION DETAILS

In this section we describe important details of our algorithm which are not a part of its theoretical core but have an impact on its performance. We also set the values of the parameters introduced throughout the paper and explain how they affect the motion estimates.

A. Gradient Approximation

Typically, a fixed discrete formula is employed to approximate scan or image gradients. In the case of range data, this strategy leads to very high values of the gradients at the object borders, which do not represent the real gradients of the observed surface(s). As an alternative, we make use of an adaptive formula that weights forward (R_θ^+) and backward (R_θ^-) derivatives in the scan with the 2D distances between contiguous observations (points):

$$R_\theta(n) = \frac{d(n+1)\, R_\theta^-(n) + d(n)\, R_\theta^+(n)}{d(n+1) + d(n)}, \tag{27}$$

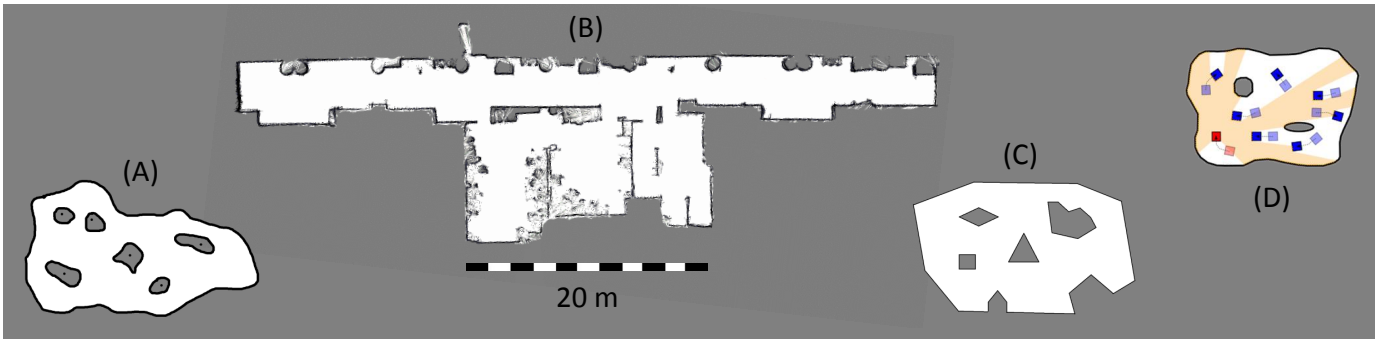

Fig. 6. Occupancy gridmaps of the environments used in the experiments (simulations). (A) Synthetic map based on curved lines, (B) real map of an office environment, (C) synthetic polygonal map and (D) dynamic environment with additional mobile robots to test robustness to moving objects.

$$d(n) = \|((x(n) - x(n-1), y(n) - y(n-1))\| \ ,$$

where n refers to a specific index in the scan. Thus, the closest neighbour always contributes more to the gradient computation while very distant points barely affect it. If both neighbours are approximately equidistant, the presented formula is equivalent to a centred finite-difference approximation.

B. Motion Filter

The environments in which the robot operates sometimes includes rooms or areas where the sensor motion cannot be fully recovered, e.g. a long corridor. This is the so-called "aperture problem": some components of the motion are undetermined and the solver can only provide an arbitrary solution for them. In order to mitigate this problem, we apply a low-pass filter in the eigenspace of the velocity $\boldsymbol{\xi}$ and use the previous estimate to constrain the underdetermined motion. First, we obtain the covariance matrix $\Sigma \in \mathbb{R}^{3\times3}$ associated with the IRLS solution of (21). Second, the eigenvalues of Σ are computed and analyzed to detect which components of the motion are undetermined and which are perfectly constrained. In eigenvector space, the velocity $\boldsymbol{\xi}_i^M$ provided by (21) is weighted with that of the previous time interval $\boldsymbol{\xi}_{i-1}$ to obtain the new filtered velocity $\boldsymbol{\xi}_i$:

$$[(1 + k_l)I + k_e E]\,\boldsymbol{\xi}_i = \boldsymbol{\xi}_i^M + (k_l I + k_e E)\,\boldsymbol{\xi}_{i-1} \,, \qquad (28)$$

where E is a diagonal matrix containing the eigenvalues and k_l, k_e are parameters of the filter. Actually, k_l imposes a constant weighting between the solution from the solver and the previous estimate while k_e defines how the eigenvalues affect the final estimate. These parameters are set to the following values:

$$k_l = 0.02\, e^{-(l-1)}, \quad k_e = 5 \times 10^3 e^{-(l-1)} \qquad (29)$$

where l is the pyramid level that ranges from 1 (coarsest) to the number of levels considered. These values provide good results in all the experiments presented in this paper but they have been obtained heuristically. As a general rule, these values could be decreased if the environment is known to be "geometrically well-constrained", and could be increased in the opposite case. Please refer to [8] for a more detailed explanation on how this filter is applied in a coarse-to-fine scheme.

C. Parameters for the Robust Optimization

There are several parameters that directly affect the optimization problem (21). On the one hand, the pre-weighting function w depends on two parameters (K_D and K_{2D}) and the sensor noise model σ_s. We do not present a formal procedure for tuning K_D and K_{2D} but rather use the values that empirically provided us with the most accurate results. Specifically, we set $K_D = 0.01$ and $K_{2D} = 2 \times 10^{-4}$. In general, higher values of K_D and K_{2D} lead to higher weights for points close to the sensor (whose coordinates and derivatives tend to be more precise) but will excessively downweight distant points which are sometimes necessary to constrain the estimated motion. For sensor noise, we set $\sigma_s = 0.02$ m for all cases, which is a representative average value of the noise found in common laser scanners used in robotics. On the other hand, the robust penalty function $F(\rho)$ (17) includes the parameter c which marks the limit between inliers and outliers (residuals higher than c lie on the flat area of $F(\rho)$ and therefore do not contribute during the optimization process). We use the median absolute deviation (MAD) of the residuals to tune c and, more specifically, we set $c = 4\,\text{MAD}\,(\rho_n)$. This is a high threshold for outlier rejection, in the sense that it keeps most of the observations as inliers and only those with noticeably high residuals will become outliers (e.g. moving objects).

IX. Experiments

We present a thorough evaluation of our method with simulated and real data in static and dynamic environments. First, we analyze the contribution of each component of our algorithm to its overall performance. We compare different versions of it: with or without pre-weighting (19), robust or non-robust minimization, symmetric versus nonsymmetric formulation, and multi-scan versus scan-to-scan alignment. Second, our method is quantitatively and qualitatively compared with some of the most prominent algorithms on scan matching.

For the synthetic experiments we use the environments presented in Fig. 6 and simulate a Hokuyo UTM-30LX laser scanner (see Table I), including Gaussian noise in the range measurements of $\sigma = 1$ cm. Note that to avoid any bias in these experiments related to the learned error model presented in Section VII, we employ different testing scenarios here.

TABLE II
EFFECT OF PRE-WEIGHTING AND ROBUST MINIMIZATION - TRANSLATIONAL AND ROTATIONAL DEVIATIONS PER SECOND, AND OVERALL DRIFT.

		Translational RMSE (cm/s)				Rotational RMSE (deg/s)				Overall drift (m)			
Map	Traj. length (m)	SQ	SQ+PW	RF	RF+PW	SQ	SQ+PW	RF	RF+PW	SQ	SQ+PW	RF	RF+PW
A	148.3	86.87	0.688	0.396	**0.298**	7.147	0.096	0.096	**0.026**	12.98	0.799	0.277	**0.225**
B	137.5	617.0	1.212	0.596	**0.505**	6.915	0.105	0.069	**0.044**	483.4	0.862	**0.156**	0.164
C	144.9	193.4	0.759	0.436	**0.225**	15.06	0.129	0.104	**0.023**	42.35	0.201	0.264	**0.084**

In both the real and simulated experiments, we do not make use of the robot wheel odometry as an initial guess for the motion estimate. Unlike other approaches, which assume that a good initial estimate is always provided, we consider that the scans are the only available inputs for the algorithms.

All the experiments presented here have been run under Ubuntu 16.04 using a single core of an Intel(R) Core(TM) i7-2600K at 3.40GHz.

A. Comparative Analysis of Each Component of the Algorithm

In this section, several versions of our algorithm are evaluated. We simulate a robot equipped with a laser scanner navigating randomly around the free space of the maps shown in Fig. 6. We use simulations instead of real data because they provide a perfect ground truth, which is necessary for quantitative comparisons. The maximum translational and rotational velocities of the robot are set to 0.5 m/s and 45 degrees/s respectively.

1) Pre-Weighting and Robust Minimization: We assess the usefulness of the pre-weighting strategy (19) and the robust minimization of the residuals (16) within the overall motion estimation process. To this end, we compare four basic versions of our method which minimize:

- Squared residuals $|\rho(\boldsymbol{\xi})|^2$ without pre-weighting (SQ).
- Squared residuals $|\rho^w(\boldsymbol{\xi})|^2$ with pre-weighting (SQ+PW).
- Robust residuals $F(\rho(\boldsymbol{\xi}))$ without pre-weighting (RF).
- Robust residuals $F(\rho^w(\boldsymbol{\xi}))$ with pre-weighting (RF+PW).

In these experiments, the motion is estimated by aligning consecutive scans (we do not evaluate the multi-scan formulation yet). We simulate a robot navigating for 10 minutes in the three environments A-C shown in Fig. 6. The scanning frequency is set to 5 Hz. The estimation errors are measured as the root mean square (RMSE) translational and rotational deviations per second, as described in [42]. Results are presented in Table II, where it can be seen that all versions except SQ provide fairly good estimates with overall translational drifts always far below 1% of the distance travelled by the robot. Robust estimation without pre-weights is more accurate than non-robust estimation with pre-weights, but it is the combination of both strategies which leads to the best results. Moreover, the associated runtimes (Table III) show that the pre-weighting actually accelerates convergence of the solver when combined with the robust function $F(\rho)$.

2) Symmetric vs Non-symmetric Formulation: In this section we compare two different versions of our method: one based on symmetric range flow (5) (SRF) as described in Section III, and another derived from the standard non-symmetric range flow constraint (2) (NSRF). For this and the rest of the experiments presented below we minimize the geometric residuals using pre-weighting and the robust penalty function $F(\rho)$. The methodology is similar to that described in Section IX-A1 but in this case two different scanning frequencies are employed: 2Hz and 5Hz.

TABLE III
RUNTIME (MS) OF THE SIMPLIFIED VERSIONS OF OUR ALGORITHM.

SQ	SQ+PW	RF	RF+PW
0.629	0.645	1.254	1.139

TABLE IV
SYMMETRIC VS NONSYMMETRIC FORMULATION - TRANSLATIONAL AND ROTATIONAL DEVIATIONS PER SECOND, AND OVERALL DRIFT.

Map	Traj. length (m)	Translational RMSE (cm/s)		Rotational RMSE (deg/s)		Overall drift (m)	
		SRF	NSRF	SRF	NSRF	SRF	NSRF
A (5Hz)	158.0	**0.314**	0.323	**0.025**	**0.025**	0.11	**0.01**
B (5Hz)	122.8	**0.517**	0.537	**0.043**	0.048	0.33	**0.29**
C (5Hz)	163.9	**0.227**	0.231	**0.025**	**0.025**	**0.21**	0.23
A (2Hz)	374.6	**0.233**	0.325	**0.016**	0.274	**0.15**	2.27
B (2Hz)	376.9	**0.704**	1.433	**0.736**	1.293	**2.55**	3.19
C (2Hz)	393.1	**0.191**	0.372	**0.017**	0.219	**0.27**	2.08

Results are shown in Table IV. It can be seen that differences between the two methods are negligible when the scanning frequency is 5Hz, but they become significant for 2Hz. These results are consistent with the theory presented in Section III. When the scanning frequency is 5Hz consecutive scans are close to each other and hence both linear approximations (2) and (5) are valid and provide accurate results. However, the alignment of consecutive scans taken at 2Hz involves estimating larger translations and rotations for which non-symmetric linearization (2) is no longer valid.

3) Scan-to-Scan vs Multi-Scan Alignment: In this section we compare three different strategies to estimate motion: consecutive-scan alignment (CA), keyscan-based alignment (KA) and the multi-scan approach (MA) described in Section V. Experiments include normal operation at 5Hz in static and dynamic environments, the estimation of large displacements when the laser frequency is set to 2Hz and estimation with noisy measurements. For the experiment with moving objects, we introduce additional robots in the simulation that are permanently wandering and contradict the assumption of a "static environment". In this case we employ a very simple synthetic map (Fig. 6-D) which, due to its limited number of obstacles and small dimensions, ensures that the additional robots are visible from the laser scanner. As a consequence of the multiple random navigations, which often lead to some robots blocking the way of the others, the distance travelled during these tests is considerably shorter than in the remainder

TABLE V
DIFFERENT SCAN ALIGNMENT STRATEGIES - TRANSLATIONAL AND ROTATIONAL DEVIATIONS PER SECOND, AND OVERALL DRIFT.

Experiment	Map	Trajectory length (m)	Translational RMSE (cm/s)			Rotational RMSE (deg/s)			Overall drift (m)		
			CA	KA	MA	CA	KA	MA	CA	KA	MA
Standard (5Hz)	A	162.6	0.327	**0.208**	0.222	0.025	0.018	**0.017**	0.152	0.045	**0.034**
	B	168.7	0.596	**0.421**	0.452	0.043	0.032	**0.030**	0.595	**0.315**	0.433
	C	165.4	0.252	**0.182**	0.186	0.024	**0.018**	**0.018**	0.092	**0.033**	**0.033**
Large displ. (2Hz)	A	175.8	0.216	**0.181**	0.186	0.018	0.016	**0.015**	0.320	0.140	**0.101**
	B	139.7	0.905	0.938	**0.734**	0.614	0.483	**0.360**	**1.417**	1.959	2.646
	C	152.8	0.193	0.178	**0.173**	0.017	0.017	**0.015**	0.017	0.020	**0.015**
noise (σ = 0.1m, 5Hz)	A	172.9	1.894	1.557	**1.397**	0.205	0.151	**0.141**	0.372	0.766	**0.256**
	B	163.8	1.922	1.613	**1.555**	0.095	0.096	**0.078**	0.502	0.416	**0.302**
	C	148.9	1.803	1.542	**1.418**	0.203	0.149	**0.139**	1.386	0.590	**0.496**
moving objects (5Hz)	D (3 obj.)	17.65	0.251	0.363	**0.186**	0.046	0.063	**0.037**	0.123	**0.097**	0.104
	D (5 obj.)	16.34	0.364	0.943	**0.197**	0.051	0.428	**0.031**	0.524	0.080	**0.076**
	D (7 obj.)	13.94	0.273	0.616	**0.261**	**0.061**	0.184	**0.061**	0.276	0.083	**0.062**

of the experiments.

Results are presented in Table V. It can be observed that KA and MA provide equally good results in static environments when the scanning frequency is 5Hz, while CA always being less accurate both locally and globally. However, the multi-scan formulation is the most precise alternative in the other (more challenging) experiments. It is slightly more precise than KA for large displacements because it is less dependent on the right selection of keyscans. It is also more precise when tested with very noisy scans (σ = 0.1m) since it has more information than CA or KA alone with which to constrain the motion estimate. Lastly, it significantly outperforms CA and KA in the presence of moving objects: the use of multiple scans implicitly facilitates the discernment and downweighting of the moving parts of the scene. In this case, MA is locally as smooth as CA (KA provides trembling estimates which lead to high relative errors) and overall precise as KA.

B. Comparisons with Other Methods

In this section we compare our approach (SRF) with the Polar Scan Matcher (PSM) [11], the Canonical Scan Matcher (CSM) [10], the Normal Distributions Transform (NDT) [12] and the Signed Distance Function-based SLAM (SDF) [13]. We consider here two versions of SRF, a complete version based on symmetric range flow and multi-scan alignment, and a trimmed-down version (T-SRF) disabling the motion filter described in Section VIII-B. For PSM, CSM and SDF we use the original code published by the authors, while for NDT we employ the implementation available in the Point Cloud Library [43][2].

1) Simulation experiments: We provide a general quantitative evaluation of the different methods in simulation, where ground truth is available. Firstly, we evaluate their performances in static environments (Fig. 6 A-C) with two scanning frequencies (5Hz and 2Hz). Secondly, we evaluate the accuracy of all methods in the presence of moving objects (map D in Fig. 6), i.e. when assumption (11) is violated. Thirdly we include an additional test in an outdoor environment. The latter case is unfavourable for our approach, which performs dense alignment without explicit correspondences and therefore requires piece-wise differentiable scans to work. The map of the Freiburg Building dataset [40] is used for this test, which is approximately 250×250 meters wide and contains large empty areas and many scattered points.

Results are shown in Table VII. Our method clearly outperforms PSM, NDT and SDF for both small (5Hz) and large (2Hz) motions, CSM being the only close competitor with good relative errors on average. In general, PSM and SDF perform poorly in all sequences, requiring higher scan frequencies than the ones tested (i.e less displacement between scans) to obtain good estimates. This is specially noticeable for SDF, which often fails to register new scans to its internal map/representation of the environment (even at 5Hz), with consequent errors in the odometry estimation and map updating. NDT is the third best candidate after SRF and CSM. It estimates translations fairly well but copes poorly with large rotations and scans with many sparse points (see outdoor results). It exhibits reasonably good results in the presence of moving objects, with error values similar to those for CSM. As can be seen from Table VII, our approach still outperforms the other methods in the outdoor test with an RMSE improvement of more than 50% relative to its closest competitor (CSM). Although the scans were much sparser in this test than in the other experiments, the structure of some buildings was often visible from the lidar (otherwise our method would not have been able to align the scans and CSM would have provided the best results). As expected the performance of the trimmed version without motion filter (T-SRF) is slightly worse than that of the complete SRF version, while still outperforming the other methods.

The average runtimes shown in Table VI illustrate that our method is not only more accurate but also much faster than the rest of the compared methods, which demonstrates its superiority to current state-of-the-art scan matchers.

TABLE VI
RUNTIME (MS) OF THE DIFFERENT METHODS.

SRF	CSM	PSM	NDT	SDF
1.617	10.187	3.947	124.1	12.649

2) Comparisons with Real Data: In this section we evaluate the different methods with real datasets, and present qualitative and quantitative results. Specifically, we utilize two of the

[2]The original code should provide better results with a lower runtime, as supported by [12], but unfortunately it is not publicly available.

TABLE VII
GENERAL COMPARISON OF OUR APPROACH WITH OTHER METHODS - TRANSLATIONAL AND ROTATIONAL DEVIATIONS PER SECOND.

	Translational RMSE (cm/s)						Rotational RMSE (deg/s)					
Map	SRF	T-SRF	CSM	PSM	NDT	SDF	SRF	T-SRF	CSM	PSM	NDT	SDF
A (5Hz)	**0.208**	0.236	0.826	11.991	2.868	7.802	**0.017**	0.020	0.156	1.632	1.095	1.122
B (5Hz)	**0.374**	0.408	0.958	2.039	3.572	23.17	**0.028**	0.029	0.073	0.732	1.298	3.644
C (5Hz)	**0.166**	0.177	0.468	11.03	2.793	10.75	**0.016**	0.017	0.074	0.920	0.756	2.278
A (2Hz)	**0.192**	0.209	2.311	12.72	10.91	12.91	**0.012**	0.019	0.320	3.852	6.454	3.661
B (2Hz)	**0.409**	0.570	4.804	12.70	14.76	22.41	**0.480**	0.522	1.556	5.282	7.709	7.449
C (2Hz)	**0.169**	0.267	2.258	13.97	11.23	20.73	**0.016**	0.051	0.123	3.802	4.493	6.704
D (3 obj.)	**0.223**	0.246	0.701	9.844	1.226	7.714	**0.038**	0.052	0.207	2.974	0.724	2.328
D (5 obj.)	**0.360**	0.382	0.914	3.474	1.457	6.012	**0.050**	0.076	0.151	0.665	1.447	2.853
D (7 obj.)	**0.264**	0.381	0.549	3.480	1.138	4.730	**0.059**	0.103	0.122	1.436	1.134	2.471
Outdoor	**0.426**	0.798	0.996	4.629	14.99	20.52	**0.028**	0.308	0.354	0.675	6.085	9.275

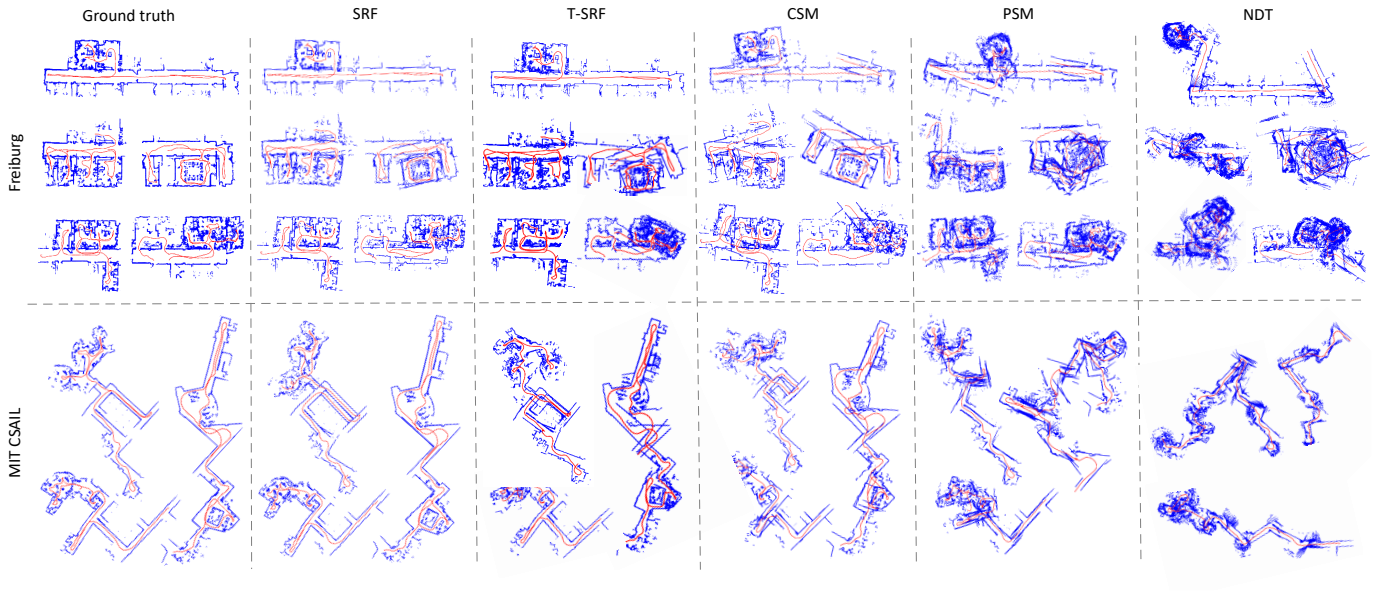

Fig. 7. Sets of Sub-maps built from the Freiburg and MIT CSAIL datasets. The scan measurements are shown as blue point clouds and the trajectories are plotted in red.

datasets listed in [44]: the *Freiburg indoor building 079* and the *MIT CSAIL building* (introduced in [45]). These datasets are conceived to test SLAM algorithms and cover long distances (423 and 380 meters), very often re-visiting the same place one or more times to test loop-closure strategies. Moreover, consecutive scans are not very close to each other, which makes them particularly challenging for methods based on pure incremental odometry. Given this complexity and the low performance demonstrated by SDF in Section IX-B1 (Table VII), we excluded it from this comparison.

One important drawback of these datasets is that no ground truth for the robot pose is provided. Instead, they include a precise trajectory estimated with SLAM algorithms, which can be used for quantitative evaluation in some applications. For the odometry estimation problem at hand, only the Freiburg dataset provides a suitable estimated trajectory that can be used as ground truth. In the case of the MIT CSAIL dataset, the laser/pose data is too decimated and therefore insufficient to evaluate odometry. As a consequence, we present quantitative and qualitative results for the Freiburg dataset, but only qualitative results for the MIT CSAIL dataset.

Qualitative results are obtained by fusing the scan observations in a 2D map according to the estimated trajectories provided by each method. Since the datasets are long and the same places are often re-visited, we divide the sequences into a few sub-sequences and build the corresponding sub-maps instead of just one large map per sequence. The resulting estimated maps, together with the maps built from the ground truth, are depicted in Fig. 7. It can be seen that our approach achieves the best results in both datasets and for every sub-sequence. For the MIT CSAIL dataset, it estimates an almost perfect trajectory for 2 of the 3 sub-maps (bottom-left and right) using only odometry in a sequence which would normally require global pose optimization and loop closure detection. Finally, the comparison between the full and trimmed versions of our approach highlights the contribution of the motion filter when facing challenging environments. The estimated trajectories with T-SRF accumulate more errors and therefore the reconstructed sub-maps are less precise. Yet, for many of the sub-sequences T-SRF still outperforms the other scan matchers.

Quantitative results are obtained by measuring the RMS

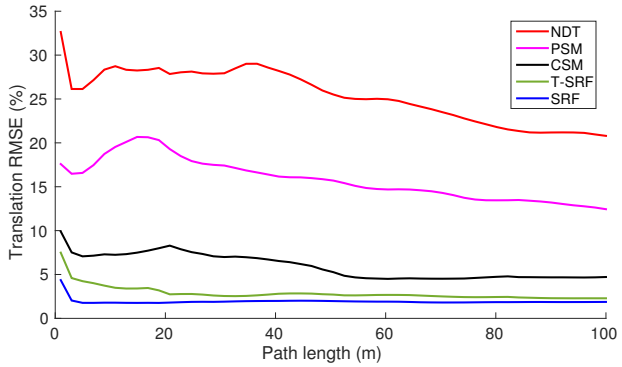

Fig. 8. Translational RMS errors per segment length computed for the Freiburg dataset, using path lengths from 1 to 100 metres. The errors are shown as a percentage of the segment length under consideration, which illustrates the drift to be expected if trajectories of the given length are to be estimated.

translational errors per segment length, as described in [46]. We compute these errors for different segment lengths ranging from 1 to 100 meters, and show them in Fig. 8 as a percentage of the segment length considered. Errors for the shortest segment lengths are not very accurate because the ground truth, although globally consistent, is not exact for local pose increments. Our algorithm has the lowest drift, with RMS errors around 2%, followed by its trimmed-down version, with errors in the interval from 2% to 5%. The error associated with CSM ranges from 5% to 7.5%, while the errors associated with PSM and NDT are always above 10% and 20% respectively. These results are less precise than the ones obtained in simulations because the laser scanner employed to record the datasets has a narrower field of view and a smaller size and also because the observed environments contain a high ratio of scattered observations which complicate the scan alignment.

X. Conclusion

This paper extends and improves the work presented in [9]. We have derived the range flow constraint from a symmetric expression of geometric consistency and have incorporated a multi-scan formulation that combines the advantages of consecutive scan alignment and keyscan-based approaches. We have also described a procedure to model the average error of our algorithm for different laser scanners and have used this model to define a keyscan-update criterion as a direct function of the maximum desirable translation and rotation errors. We have presented a large set of experiments used to evaluate our method with simulated and real data and to compare it with several state-of-the-art algorithms in scan matching. Quantitative and qualitative results demonstrate that our method is significantly more accurate and faster both in static and dynamic environments.

However, results from the datasets show that the performance of our algorithm deteriorates when the laser primarily sees scattered points in the environment. This is expected because our method relies on range gradients and requires at least a piecewise-differentiable range function to align the scans. For future work, we plan to combine our dense formulation with other sparse techniques, adding an extra term to our cost function to also align interest points.

References

[1] R. Triebel, K. Arras, R. Alami, L. Beyer, S. Breuers, R. Chatila, M. Chetouani, D. Cremers, V. Evers, M. Fiore *et al.*, "Spencer: A socially aware service robot for passenger guidance and help in busy airports," in *Field and Service Robotics*, 2016, pp. 607–622.

[2] A. Orlandini, A. Kristoffersson, L. Almquist, P. Bjrkman, A. Cesta, G. Cortellessa, C. Galindo, J. González-Jiménez, K. Gustafsson, A. Kiselev, A. Loufti, F. Melendez-Fernandez *et al.*, "ExCITE Project: A review of forty-two months of robotic telepresence technology evolution," *Presence: Teleoperators and Virtual Environments*, 2017.

[3] S. Coradeschi, A. Cesta, G. Cortellessa, L. Coraci, C. Galindo, J. Gonzlez-Jimnez, L. Karlsson *et al.*, *GiraffPlus: A System for Monitoring Activities and Physiological Parameters and Promoting Social Interaction for Elderly*, ser. Human-Computer Systems Interaction: Backgrounds and Applications 3. Springer, 2014, pp. 261–271.

[4] E. Ackerman, "Fetch robotics introduces Fetch and Freight: Your warehouse is now automated," IEEE Spectrum online, 2015.

[5] F. Steinbrücker, J. Sturm, and D. Cremers, "Real-time visual odometry from dense RGB-D images," in *Proc. Int. Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 719–722.

[6] T. Tykkälä, C. Audras, and A. I. Comport, "Direct iterative closest point for real-time visual odometry," in *Proc. Int. Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 2050–2056.

[7] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for RGB-D cameras," in *IEEE Int. Conference on Robotics and Automation (ICRA)*, 2013, pp. 3748–3754.

[8] M. Jaimez and J. Gonzalez-Jimenez, "Fast visual odometry for 3-D range sensors," *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 809 – 822, 2015.

[9] M. Jaimez, J. G. Monroy, and J. Gonzalez-Jimenez, "Planar odometry from a radial laser scanner. A range flow-based approach," in *IEEE Int. Conference on Robotics and Automation (ICRA)*, 2016, pp. 4479–4485.

[10] A. Censi, "An ICP variant using a point-to-line metric," in *IEEE Int. Conference on Robotics and Automation (ICRA)*, 2008, pp. 19–25.

[11] A. Diosi and L. Kleeman, "Fast laser scan matching using polar coordinates," *The International Journal of Robotics Research*, vol. 26, no. 10, pp. 1125–1153, 2007.

[12] P. Biber and W. Strasser, "The Normal Distributions Transform: A new approach to laser scan matching," in *IEEE Int. Conference on Intelligent Robots and Systems (IROS)*, 2003, pp. 2743–2748.

[13] J. D. Fossel, K. Tuyls, and J. Sturm, "2D-SDF-SLAM: A signed distance function based SLAM frontend for laser scanners," in *IEEE Int. Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 1949–1955.

[14] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo localization for mobile robots," *Artificial intelligence*, vol. 128, no. 1, pp. 99–141, 2001.

[15] J. L. Blanco, J. Gonzalez, and J. A. Fernandez-Madrigal, "Optimal filtering for non-parametric observation models: applications to localization and SLAM," *The International Journal of Robotics Research*, vol. 29, no. 14, pp. 1726–1742, dec 2010.

[16] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[17] J. Gonzalez and R. Gutierrez, "Direct motion estimation from a range scan sequence," *Journal of Robotic Systems*, vol. 16, no. 2, pp. 73–80, 1999.

[18] G. D. Tipaldi and K. O. Arras, "FLIRT - Interest regions for 2D range data," in *IEEE Int. Conference on Robotics and Automation (ICRA)*, 2010, pp. 3616–3622.

[19] F. Kallasi, D. L. Rizzini, and S. Caselli, "Fast keypoint features from laser scanner for robot localization and mapping," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 176–183, 2016.

[20] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Robotics-DL tentative*, 1992, pp. 586–606.

[21] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2D range scans," *Journal of Intelligent and Robotic Systems*, vol. 18, no. 3, pp. 249–275, 1997.

[22] J. Minguez, L. Montesano, and F. Lamiraux, "Metric-based iterative closest point scan matching for sensor displacement estimation," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 1047–1054, 2006.

[23] J. Gonzalez, A. Stentz, and A. Ollero, "A mobile robot iconic position estimator using a radial laser scanner," *Journal of Intelligent and Robotic Systems*, vol. 13, no. 2, pp. 161–179, 1995.
[24] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," in *Proc. of Robotics: Science and Systems (RSS)*, 2009.
[25] A. W. Fitzgibbon, "Robust registration of 2D and 3D point sets," *Image and Vision Computing*, vol. 21, no. 13, pp. 1145–1153, 2003.
[26] A. Censi, L. Iocchi, and G. Grisetti, "Scan matching in the Hough domain," in *IEEE Int. Conference on Robotics and Automation (ICRA)*, 2005, pp. 2739–2744.
[27] A. Diosi and L. Kleeman, "Laser scan matching in polar coordinates with application to SLAM," in *IEEE Int. Conference on Intelligent Robots and Systems (IROS)*, 2005, pp. 3317 – 3322.
[28] E. B. Olson, "Real-time correlative scan matching," in *IEEE Int. Conference on Robotics and Automation (ICRA)*, 2009, pp. 4387–4393.
[29] M. Achtelik, A. Bachrach, R. He, S. Prentice, and N. Roy, "Stereo vision and laser odometry for autonomous helicopters in GPS-denied indoor environments," in *SPIE Unmanned Systems Technology XI*, 2009, pp. 1–10.
[30] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grixa, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 46–56, 2012.
[31] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP variants on real-world data sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, 2013.
[32] H. Spies, B. Jahne, and J. L. Barron, "Range flow estimation," *Computer Vision and Image Understanding*, vol. 85, no. 3, pp. 209–231, 2002.
[33] A. Censi and R. M. Murray, "Bootstrapping bilinear models of simple vehicles," *The International Journal of Robotics Research*, vol. 34, no. 8, pp. 1087–1113, 2015.
[34] L. Alvarez, C. A. Castano, M. Garcia, K. Krissian, L. Mazorra, A. Salgado, and J. Sanchez, "Symmetric optical flow," in *Int. Conference on Computer Aided Systems Theory*, 2007, pp. 676–683.
[35] R. Battiti, E. Amaldi, and C. Koch, "Computing optical flow across multiple scales: an adaptive coarse-to-fine strategy," *International Journal of Computer Vision*, vol. 6, no. 2, pp. 133–145, 1991.
[36] C. Zach, "Robust bundle adjustment revisited," in *European Conference on Computer Vision (ECCV)*, 2014, pp. 772–787.
[37] J. Stückler and S. Behnke, "Integrating depth and color cues for dense multi-resolution scene mapping using RGB-D cameras," in *IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2012, pp. 162–167.
[38] M. Meilland, A. I. Comport, and P. Rives, "Dense visual mapping of large scale environments for real-time localisation," in *IEEE Int. Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 4242–4248.
[39] C. Kerl, J. Sturm, and D. Cremers, "Dense visual SLAM for RGB-D cameras," in *IEEE Int. Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 2100–2106.
[40] A. Howard and N. Roy, "The robotics data set repository (radish)," 2003. [Online]. Available: http://radish.sourceforge.net/
[41] J. R. Ruiz-Sarmiento, C. Galindo, and J. Gonzalez-Jimenez, "Robot@home, a robotic dataset for semantic mapping of home environments," *International Journal of Robotics Research*, vol. 36, no. 2, pp. 131 – 141, 2017.
[42] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IEEE Int. Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 573–580.
[43] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *IEEE Int. Conference on Robotics and Automation (ICRA)*, 2011, pp. 1–4.
[44] C. Stachniss, "Robotics datasets." [Online]. Available: http://www2.informatik.uni-freiburg.de/~stachnis/datasets.html
[45] C. Stachniss, G. Grisetti, W. Burgard, and N. Roy, "Analyzing Gaussian proposal distributions for mapping with rao-blackwellized particle filters," in *IEEE Int. Conference on Intelligent Robots and Systems (IROS)*, 2007, pp. 3485–3490.
[46] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proc. Int. Conference on Computer Vision (ICCV)*, 2012, pp. 3354–3361.

Mariano Jaimez is a PhD student associated with the Machine Perception and Intelligent Robotics group (MAPIR) at the University of Málaga and with the Computer Vision group at the Technical University of Munich. He received his B.S. and M.S. degrees in Mechatronics from the University of Málaga (Spain) in 2010 and 2012 respectively. His research interests encompass the potential applications of range-sensing technologies in the fields of robotics, computer vision, virtual/augmented reality and autonomous driving. His latest works focus on visual odometry, scene flow estimation and 3D reconstruction.

Javier Monroy is a postdoc researcher associated with the Machine Perception and Intelligent Robotics group (MAPIR) at the University of Málaga (Spain). He received his B.Sc (2007), and M.Sc (2010) in Electrical Engineering from this University, and his Ph.D, focussing on mobile robotics perception, in 2013. During spring 2011 he was a guest researcher at the Applied Autonomous Sensor Systems, Örebro University (Sweden), collaborating with the Mobile Robotics and Olfaction group. His research interests include mobile robotics, machine learning and electronic noses. In these fields, he has authored more than 15 scientific works.

Manuel Lopez-Antequera is a PhD student associated with the Machine Perception and Intelligent Robotics group (MAPIR) at the University of Málaga and the Intelligent Systems group at the Johann Bernoulli Institute for Mathematics and Computer Science in the University of Groningen. He received his B.Sc (2009), and M.Sc (2012) in Electrical Engineering from the University of Málaga. His research interests include robot localization and mapping, machine learning and computer vision.

Javier Gonzalez-Jimenez is the head of the MAPIR group and full professor at the University of Málaga. Prof. Gonzalez-Jimenez received his B.S. degree in Electrical Engineering from the University of Seville in 1987. He joined the Department of "Ingenieria de Sistemas y Automatica" at the University of Málaga in 1988 and received his Ph.D. from this University in 1993. In 1990-1991 he was at the Field Robotics Center, Robotics Institute, Carnegie Mellon University (USA) working on mobile robots as part of his PhD. Since 1996 he has been heading Spanish and European projects on mobile robotics and perception. In these areas he is (co)author of more than 200 scientific papers.