Auton Robot (2009) 27: 353–371
DOI 10.1007/s10514-009-9156-5

Rawseeds ground truth collection systems for indoor self-localization and mapping

Simone Ceriani · Giulio Fontana · Alessandro Giusti · Daniele Marzorati · Matteo Matteucci · Davide Migliore · Davide Rizzi · Domenico G. Sorrenti · Pierluigi Taddei

Received: 13 February 2009 / Accepted: 1 September 2009 / Published online: 22 September 2009

Abstract A trustable and accurate ground truth is a key requirement for benchmarking self-localization and mapping algorithms; on the other hand, collection of ground truth is a complex and daunting task, and its validation is a challenging issue. In this paper we propose two techniques for indoor ground truth collection, developed in the framework of the European project RAWSEEDS, which are mutually independent and also independent on the sensors onboard the robot. These techniques are based, respectively, on a network of fixed cameras, and on a network of fixed laser scanners. We show how these systems are implemented and deployed, and, most importantly, we evaluate their performance; moreover, we investigate the possible fusion of their outputs.

Keywords Ground truth · Benchmarking · Pose estimation · Mobile robotics

This work has been supported by the European Commission, Sixth Framework Programme, Information Society Technologies, Contract Number FP6-045144 (RAWSEEDS).

S. Ceriani · G. Fontana · A. Giusti · M. Matteucci (✉) · D. Migliore · D. Rizzi · P. Taddei
Dipartimento di Elettronica ed Informazione, Politecnico di Milano, Milan, Italy
e-mail: matteucci@elet.polimi.it

S. Ceriani
e-mail: ceriani@elet.polimi.it

G. Fontana
e-mail: fontana@elet.polimi.it

A. Giusti
e-mail: giusti@elet.polimi.it

D. Migliore
e-mail: migliore@elet.polimi.it

P. Taddei
e-mail: taddei@elet.polimi.it

D. Marzorati · D.G. Sorrenti
Dip. di Informatica, Sistemistica e Comunicazione, Università di Milano—Bicocca, Milan, Italy

D. Marzorati
e-mail: marzorati@disco.unimib.it

D.G. Sorrenti
e-mail: sorrenti@disco.unimib.it

1 Introduction

Progress in the field of robotics requires that robots gain the ability to operate with less and less direct human control, without detriment to their performance and, most importantly, to the safety of the people interacting with them. A key factor for progress in this field is a substantial advancement in the performance of robots associated to the concept of autonomy. Among the many facets of autonomy, moving safely in the environment, and being able to reach a goal location is the basic ability that a robot must necessarily possess. This requires, in particular, the robot to be capable to localize itself in the environment; this is usually done by building some form of internal representation of the environment, i.e., a map, and locating the positions of the robot and its goal on the map.

Any mobile autonomous robot must have the abilities needed to perform activities of mapping, self-localization or Simultaneous Localization And Mapping (SLAM) (Thrun et al. 2004, 2005; Folkesson and Christensen 2004; Lu and Milios 1997). Obviously, these abilities are not sufficient to ensure that the robot is also able to execute a task, but they can be thought of as necessary conditions for a mobile robot to be capable of effective autonomous behavior. There are a

few applicative cases where the world modeling functionality is not required, e.g., when the map is known and static. In common applications, the mapping and self-localization functionalities are usually tackled simultaneously, so to integrate the information gathered during the exploration.

The main problem that SLAM algorithms have to face is perhaps the fact that the sensor data are affected by imperfections, like noise, limited dynamic range, systematic errors, etc. These imperfections are significant for all sensor, but they are even more critical with low-cost sensors, whose use is unavoidable in most mass-market robotic domains, because of economic constraints: "*extensive market analyses show that a complex sensing system for a mobile robot cannot cost more that 10 US$, for a consumer-level robot*" (Angle 2004). Pirjanian (2007) was less strict, but the mentioned budget, in the order of tens of US$ including processing power, was still quite far from the current budgets in sensors for mobile robotics. The algorithms for solving the mapping and localization problem become much more complex when multiple sensors are used (as is usually done to partially compensate for the intrinsic limitations of each sensor), because they need to include a process of sensor fusion between data coming from different sensors (Durrant-Whyte 1987; Tardós and Castellanos 1999).

Successful implementation, tuning and deployment of complex algorithms strictly depends on the ability to perform accurate benchmarking; the role of benchmarks is to allow testing, evaluation and comparison of existing algorithms. In turn, the availability of an accurate ground truth (GT) is a key issue in benchmarking self-localization and mapping algorithms. In particular, two set of unknowns are relevant for the autonomous navigation tasks: those representing the map, and those representing the robot pose. In indoor scenarios, the former unknowns are often easily defined by using executive drawings of the environments, while the latter might require ad-hoc solutions. This paper is focused on gathering the ground truth for the robot pose, which is usually the most complex task.

Let us consider such task as an estimation problem, i.e., to provide an estimate of the actual poses passed by the robot during its motion in the environment: then, GT is a set of very good approximations of the actual values of the unknowns, trusted by all research groups, which is a key to allow a fair performance comparison between algorithms. In order to be unbiased and act as an independent reference, GT estimates should be derived by data from different sensors than the robot's own; otherwise such GT data could not be considered as suitable for comparing algorithms that use the robot's sensors, and therefore would not be trustable. In other words, we believe we need a statistically independent measuring system, for measuring the mentioned unknowns.

The only option we found feasible, with respect to such requirement, was based on observing the robot from outside. This was also required by the fact that adding new sensors dedicated to GT onboard the robot was not feasible in the case of the RAWSEEDS mobile base; the capabilities of the robot were actually fully exploited both in power consumption and space by the sensor suite required by the project, see also Fig. 2; for further details on the data gathering system, see Sorrenti and Matteucci (2008). Our solution has been to structure the robot so that it could perceive its surroundings with a multiplicity of sensors, and then rely on a different set of sensors, distributed in the environment, to observe the robot for the GT. This solution contrasts with those that rely on robot-centric perception, such as scan matching approaches.

In this paper we present, compare and evaluate two different systems for indoor Ground Truth acquisition developed within the EC-funded RAWSEEDS project, which meet the aforementioned independency requirements:

GTvision, based on data from an external network of calibrated fixed cameras, also mentioned as *camera network* hereafter; the robot pose is reconstructed by basing on the observation, by the camera network, of a set of visual markers attached to the robot;

GTlaser, based on readings from a network of fixed laser scanners; the robot pose is reconstructed by basing on a rectangular hull attached around the robot.

We review related works in Sect. 2, then briefly present the RAWSEEDS project in Sect. 3 and discuss the experimental setup and the hardware we used for ground truth gathering (Sect. 4). The GTvision and GTlaser techniques are described in Sects. 5 and 6, respectively. In Sect. 7 we show results and performance comparisons with respect to a set of validation data, and also explore the possibility of fusing the results of GTvision and GTlaser techniques into a single improved estimate. Finally, we present conclusions in Sect. 8.

2 Related works

The procurement of a reliable and accurate Ground Truth is a key factor for the acceptance of datasets for benchmarking: we feel this is an important missing part in Radish (Howard and Roy 2003), which is currently considered the state-of-the-art in supporting comparison of algorithms. Radish is a community initiative, and a repository of datasets, provided on a voluntary base by research groups. Unfortunately, the datasets are not provided with Ground Truth, whose procurement is certainly not a trivial task; the usage of GT-less datasets is implicitly limiting the comparison to what a human could infer from the dataset itself.

In principle, a possible solution to indoor ground truth acquisition could be an off-the-shelf commercial system such as the Ultra Wide Band system developed by UBISENSE

(Ubisense 2006). We have experience of such system, whose published specifications state that an accuracy between 15 cm and 30 cm should be considered valid in best-case installations. Such installations are those involving a large number of devices and a careful configuration by Ubisense's own personnel. These requirements are often incompatible with the budgets of research labs, especially when installations in different locations are planned; moreover, the foreseen accuracy is less than WHAT we WOULD expect, for indoor data.

Several other possible solutions are based on an upward-looking camera mounted on the robot. Under this class we have methods basing on observing a marker physically placed or projected on the ceiling, possibly not in the visible range (e.g., the NorthStar by Evolution Robotics). These solutions might not be viable because (in the future) the sensing suite of the RAWSEEDS robot might include, after the successful experiences of the Rhyno and Minerva robots (Thrun et al. 2005), an upward-looking camera, that would be observing such markings, independently of them being in the visible or infra-red range. This would turn into an unacceptable advantage when using such sensor stream. One could place a NIR-cutoff filter on such camera to avoid their detection but this would have implied reducing the sensitiveness of that camera, which could have required to increase the exposure, possibly leading to motion-blur or increased noise. In short, we do not want to alter the environment in a way that might be perceived by the robot sensors.

3 The RAWSEEDS benchmarking toolkit for SLAM

RAWSEEDS (Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets) is an EC-financed project aiming at addressing the need for a benchmarking toolkit, within the field of self-localization, mapping and SLAM. This is done by generating and publishing on the internet a comprehensive *Benchmarking Toolkit* which includes:

1. several high-quality multisensorial datasets, with associated ground truth, gathered by exploring real-world environments (indoor and outdoor) with a mobile robot equipped with a wide set of sensors;
2. a set of Benchmark Problems (BPs) built on such datasets, i.e., well-defined problems that also include quantitative criteria to assess solutions to them;
3. example solutions to the BPs, called Benchmark Solutions (BSs), based on state-of-the-art algorithms and evaluated according to the criteria defined by the associated BPs.

RAWSEEDS is not the only effort towards the definition of benchmarks in SLAM, the most known being the already mentioned Robotics Data Set Repository (Radish) (Howard and Roy 2003). However, RAWSEEDS' Benchmarking Toolkit tries to overcome some of the limitations (no multisensor datasets and almost no ground truth available) of these projects. The most innovative features of the Toolkit are:

- multisensoriality: each datasets is composed of multiple, time-synchronized data streams coming from several different sensor systems on board of the robot;
- the set of sensors has been chosen to be representative of both low- and high-end types of sensors ATTACH the second item of the list to the end of the first, between parentheses for mobile robotics;
- presence of ground truth: each dataset is associated to suitable ground truth data (generated with means independent from the robot sensors) in the form of trajectories recorded in predefined areas, and executive drawings of the explored space;
- high quality: the project included an explicit validation phase, aimed at ensuring the quality of the datasets;
- wide set of scenarios: the datasets have been captured in different situations (indoor, outdoor and mixed environments), with natural and artificial light, in static (i.e., no displaced objects or moving people) and dynamic conditions;
- presence of explicit evaluation metrics: each Benchmark Problem includes the methodologies needed to evaluate objectively any solution, thus allowing comparison of different solutions independently from the actual choice of implementation and/or representation.

RAWSEEDS' Benchmarking Toolkit is available through the project's website,[1] along with the documentation needed to use it and with additional documentation about the project.

One of the goals of RAWSEEDS was to provide datasets that include data from most of the sensor families actually employed in mobile robotics. Another goal was the gathering of datasets in real-world environments, covering a wide range of locations and different experimental conditions. To reach this goal, RAWSEEDS used a custom (i.e., not commercial) robotic platform called *Robocom* (see Fig. 1), built by Università di Milano—Bicocca basing on joint background with Politecnico di Milano. The mechanical structure was inspired by the smallest version of the Volksbot family (see www.volksbot.de), although all the internals, including the low-level control were especially designed for Robocom. This platform was chosen for its small overall dimensions, associated to high payload and very good maneuverability: a quality, this, most needed in cramped indoor environments. Moreover, Robocom is capable of both indoor and (limited) outdoor operation. All in all, minimization of bulk and mass notwithstanding, the complete robot used to acquire RAWSEEDS' datasets weighed almost 85 kg and measured

[1] http://www.rawseeds.org

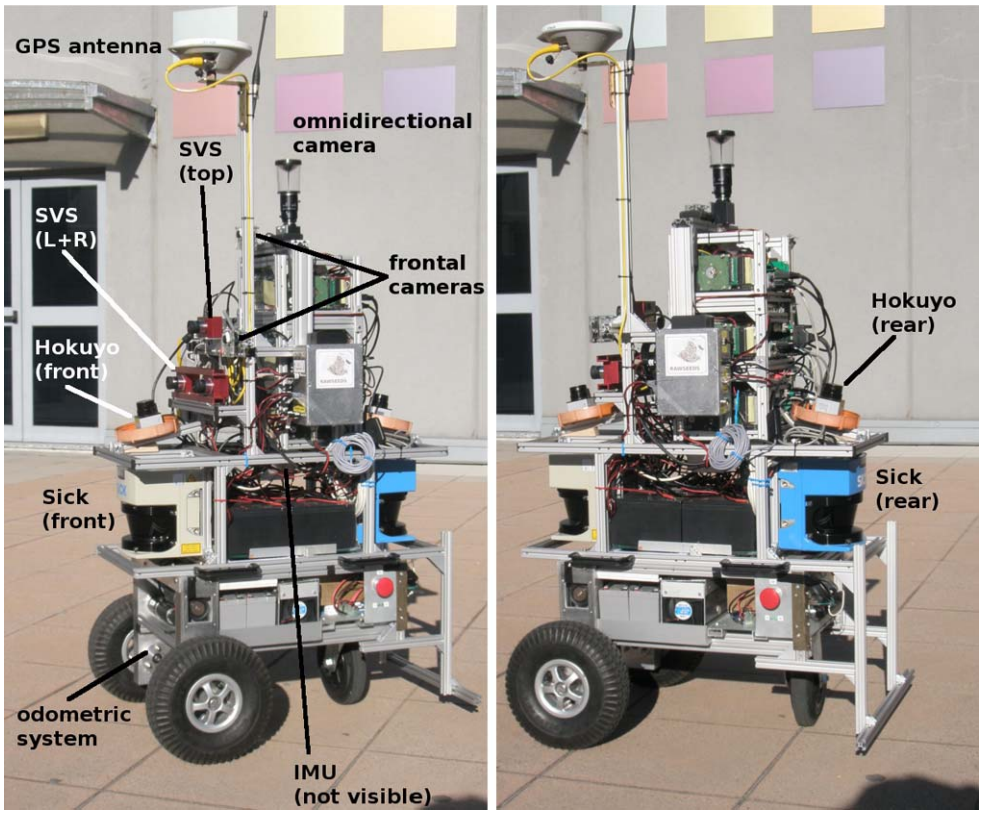


Fig. 1 Robocom platform with onboard sensors, in the outdoor version

570 (width) by 780 (length) by 1250 (height) millimeters. These data are near to the limits for easy movement in indoor environments (especially for doors and corridors), and have been cited to highlight how this aspect could easily be underestimated while designing heavily sensorized robots.

The sensor systems used by RAWSEEDS have been chosen to cover a wide range of devices, while concentrating on those which are more frequently used in the SLAM field. Sensors mounted on board of the robot are listed below.

- Odometric system fitted to the robotic base.
- Binocular vision system, composed of a b/w two-camera Videre Design STH-DCSG-VAR system.
- Trinocular vision system realized combining the binocular STH-DCSG-VAR with an additional Videre Design DCSG camera (the same camera used by the STH-DCSG-VAR).
- Color monocular vision, an Unibrain Fire-i 400 camera (chosen as representative of low-cost devices).
- Omnidirectional color vision, obtained by using a Prosilica GC1020C color GigE Vision camera fitted with an off-the-shelf hyperbolic mirror by Vstone.
- Two Hokuyo URG-04LX low-cost laser range finders, mounted on the front and the back of the robot. The scanning planes of the LRSs have been slightly tilted down towards the floor: this is coherent with a usage of such sensors as obstacle avoidance devices and slope detectors.
- Sick LMS291 and LMS200 laser range finders, mounted on the front and the back of the robot.
- Xsense MTi inertial measurement unit (version with 1.7 g full scale acceleration and 150 deg/s full scale rate of turn).
- Sonar belt composed of 12 Maxbotix EZ-2 ultrasonic emitters/receivers attached to a custom acquisition board, designed and built by Politecnico di Milano and based on a DSPIC microcontroller (not shown in the outdoor setup of Fig. 1).

In the RAWSEEDS project, we tried to include a wide set of scenarios, covering different kinds of environments. Presently, RAWSEEDS' Benchmarking Toolkit includes datasets recorded in indoor, outdoor and mixed (i.e., partially indoor and partially outdoor) locations; in natural and artificial light conditions; and in static (i.e., with no moving objects or people) or dynamic conditions. These datasets were all gathered with the same robot setup already described, the only difference being that sonar sensors were not used outdoor. In this way, the users of the datasets will also be able to verify the performance of the chosen sensor types in a range of conditions. Of course it is possible that in the future new datasets will be included into RAWSEEDS' Toolkit, possibly acquired in new locations. The locations chosen for data acquisitions are the following (for each of them, all the types of datasets available are listed between parentheses): Bicocca (indoor only, static and dynamic, artificial and

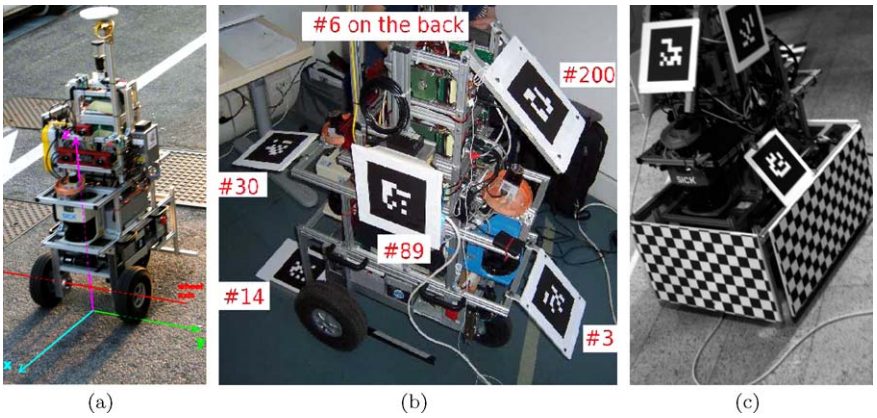


Fig. 2 The robot and its frame of reference (**a**), with visual markers (**b**) and the rectangular gown (**c**). The checkerboard on the gown is not currently exploited for visual localization

natural lighting), and Polimi-Durando (outdoor and mixed, static and dynamic, natural lighting).

Each of the datasets of RAWSEEDS' Benchmarking Toolkit is composed of data gathered by the robot on multiple paths, partially overlapping and organized into loops to help the testing of SLAM algorithms. One of the key performance elements for such algorithms is, in fact, the ability to correctly "close the loop" when the robot returns to a previously visited zone: i.e., the ability to correctly detect that the zone presently explored has been visited before and to update the map built by the algorithm accordingly. It is worth noting that each of the explored environments appears in more than one dataset, with different robot paths and/or exploring different areas, and in different conditions (e.g., static or dynamic, artificial or natural lighting). This allows for some interesting use of the datasets: e.g., simulation of multi-robot acquisition or evaluation of the map built by the SLAM algorithm on a different dataset with respect to the one used to build it.

As already mentioned, the recording of ground truth is one of RAWSEEDS' most interesting features. In outdoor environments ground truth has been collected by the use of a standard RTK-GPS apparatus with precision up to 2–6 centimeters (depending on reception of GPS satellite signals). In indoor, as pointed out in the previous section, such kind of apparatus does not exists and we had to develop the original solutions presented in this paper together with a proper validation of their accuracy. It should be noticed that ground truth covers only a subset of the indoor area covered by the robot path; however all the robot paths have been carefully designed is such a way that multiple loops are closed in the ground truth area making the most from its use.

4 Indoor ground truth definitions and hardware setup

In the RAWSEEDS project, indoor GT was gathered by equipping the robot with six planar visual markers (see Fig. 2) and a rectangular hull ("*gown*") attached to the robot's sides. Both the markers and the gown are placed in such a way to be unobstructive to the onboard sensors, and are needed for accurately reconstructing the robot position when using cameras (GTvision) and laser scanners (GT-laser), respectively.

The visual markers are a variation of the pattern commonly used in Augmented Reality, in particular in the ARToolkit system (Kato and Billinghurst 1999). Markers are black squares with a well-defined size, surrounded by a white border of variable width and including a 6×6 matrix of black and white squares, where an unique ID for each marker is encoded. Before generating GT data, the relative position of the markers on the robot hull has to be computed and we used for this a number of high-resolution images taken from a hand-held camera, as described below.

4.1 Recovering the relative positions of visual markers on the robot

In order to estimate the rigid transformations leading from the robot's frame of reference to the frame of reference of each of the six markers, we implemented an ad-hoc technique based on the ability to localize a single marker in 3D (see Sect. 5). We shot a number of images with a hand-held high-resolution camera (see Fig. 3a), whose intrinsic parameters are precisely calibrated, but whose extrinsic parameters are not known as the camera is handheld and freely moved by the operator, in order to image the robot from many different angles.

Our system analyzes each of those images in order to detect and localize each marker in 3D: for each pair of markers M_i, M_j visible in a single frame, we get two rototranslations T_i and T_j, leading from the camera to the marker's frame of reference. The relative position of M_i with respect to M_j is therefore found as $T_i^j = (T_j)^{-1} \cdot T_i$. Similarly, the inverse transformation T_j^i is also computed. Note that T_i^j and T_j^i are independent on the camera position.

We perform this operation on all the available frames, also considering the (lower-quality) data acquired with the external cameras. Finally, we consider a single marker k,

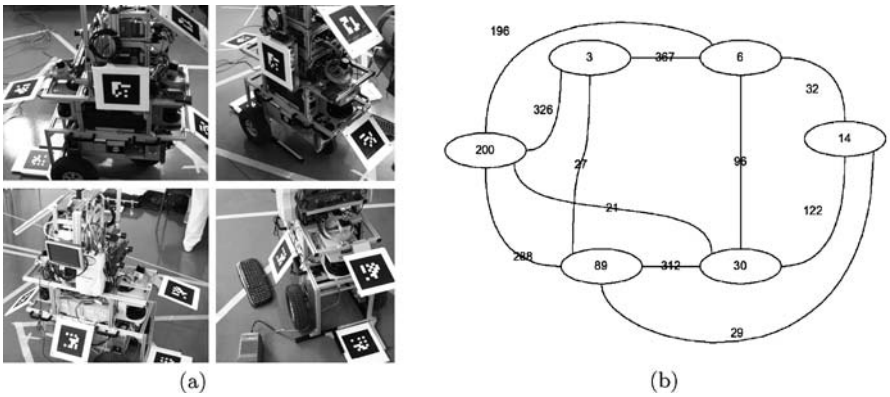


Fig. 3 Relative positions between markers: high resolution images (**a**) are processed in order to recover a number of rigid transformations between each pair of markers; the number of such relations is shown in (**b**), where nodes represent the different markers

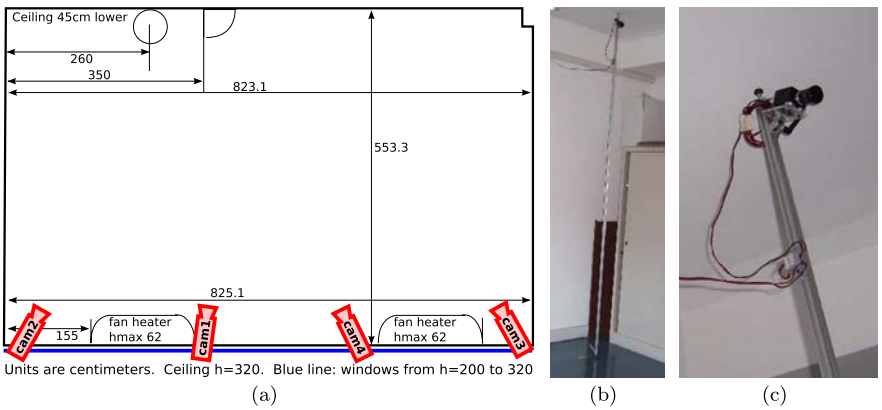


Fig. 4 Executive drawing of the environment used for initial validation of the GTvision system (see Sect. 7.1), with camera positions. Actual dataset collection was performed in a different environment

whose position in the robot's frame of reference is manually measured, and gather all the computed transformations towards each of the other markers. As some markers are not directly reachable from k (i.e., they are never detected in the same frame as k, this happens, e.g., when they lie on different sides of the robot), the composition of two rigid transformations must be computed by considering "bridge" markers. As many different rigid transformations are detected in multiple frames, for a given couple of markers (see Fig. 3b), those are then averaged in order to recover a better estimate. This operation is assisted by the user, which validates the results and ensures that no outliers are considered.

4.2 Fixed cameras

The environment is equipped with a camera network composed by a few cameras. These are mounted on poles, in order to observe the robot from above; special care has been taken in order to ensure that cameras do not move nor rotate. In order to reconstruct the robot position in the world frame of reference W, the cameras must be calibrated: i.e., their intrinsic parameters must be computed, and their position and orientation (extrinsic parameters) with respect to W must be found.

The intrinsic parameters of each camera, including radial distortion parameters, have been first calibrated by using the popular "Camera Calibration Toolbox for MATLAB" (Bouguet 2002), using a large enough chessboard, which is well visible from the cameras, see e.g., one of the two chessboards in Fig. 5. Then, the rigid transformations between the world reference frame and each of the cameras have to be computed.

Since only one camera directly images the world reference frame, and the overlapping field of view for any two adjacent cameras is quite narrow, calibrating the extrinsic camera parameters is not straightforward. In order to connect the fields of view of two adjacent cameras, we built a double calibration checkerboard mounted on a solid mechanical frame (see Fig. 5), in such a way that the relative position of the two patterns is known and stable. Then, we can place each of the two patterns in the field of view of each of the two adjacent cameras, and acquire several images with different positions and orientations of the double chessboard: this allows us to compute an accurate estimate of the relative position between the two cameras. The extrinsic parameters of each camera with respect to the "world" reference frame (which is only seen by the first camera) are finally computed by chaining the camera-to-camera rigid transformations. The accuracy of the resulting calibration data has been manually validated, and found to be satisfactory (see Sect. 7 and Sorrenti and Matteucci 2008 for numerical details).

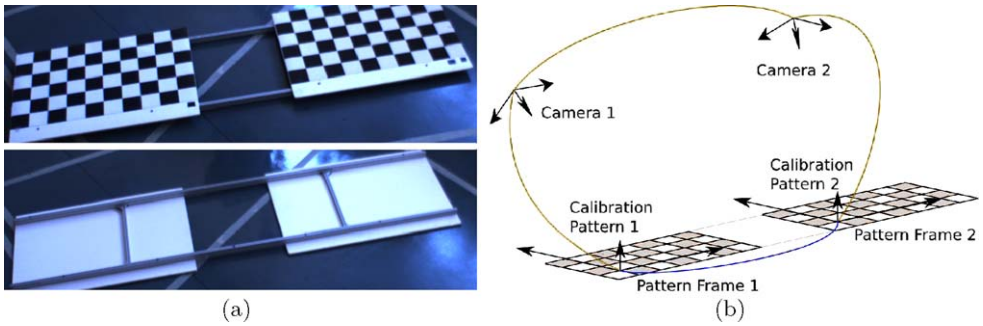


Fig. 5 (**a**) The double calibration pattern we use for calibrating cameras. (**b**) shows the involved reference systems

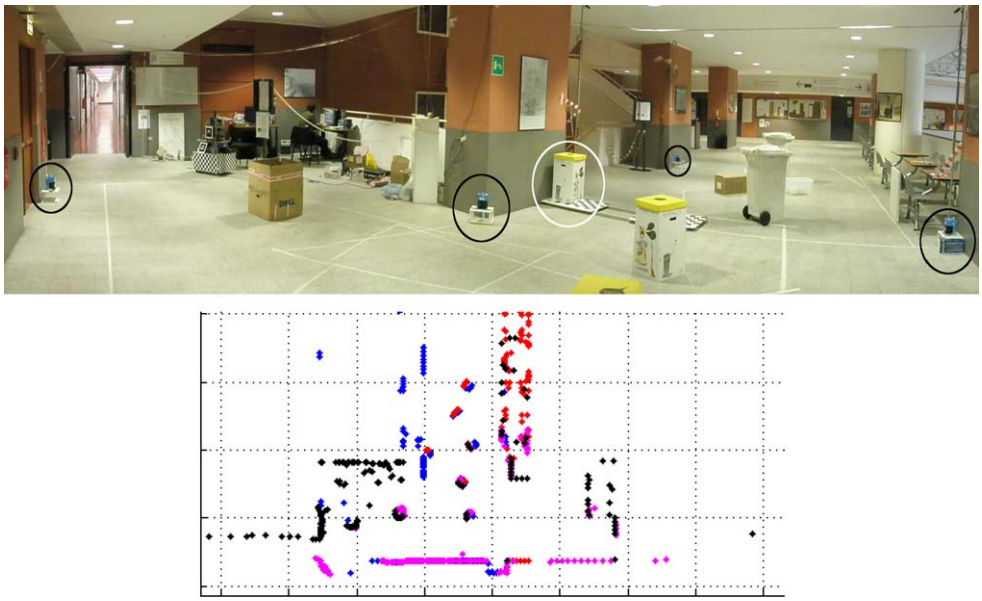


Fig. 6 *Top*: panoramic image representing the indoor environment during the laser alignment procedure. The fixed laser scanners are *circled* in black. Note the objects scattered in the room for easing alignment. The object that allows to overlap the scans with the W reference frame is *circled* in *white*; note that it features a checkerboard for extrinsic camera calibration in the same reference frame. *Bottom*: the resulting aligned scans (detail); data from the four lasers is in different colors

4.3 Fixed laser scanners

The environment was also equipped with a number of fixed planar laser scanners. Each laser scanner has a field of view of 180 deg, and returns 180 samples per scan, at 75 Hz scanning frequency. Each measurement represents the distance between the laser and the first object in that direction. The laser scanners are placed in such a way to operate on a horizontal plane set at about 25 centimeters from the ground.

We estimate the position of each laser scanner with respect to each other by aligning their output, after several objects of known shape were placed in the environment (see Fig. 6). As a result, the readings from all the laser scanners can be merged. Moreover, one of the objects has edges coinciding with the "world" reference frame. A camera calibration chessboard has been placed nearby, at a known offset from such object; this allowed us to determine a common reference frame for both cameras and laser readings.

5 Ground truth from visual markers

GTvision is computed by reconstructing the positions of visual markers attached to the robot by means of their images, acquired by the fixed calibrated cameras. As the relative positions of the markers on the robot frame are known (see Sect. 4.1) and the ID of each marker can be decoded, we can compute the position of the robot's frame of reference, even if only few (or one) markers are visible in an image.

The procedure first detects visual markers in each frame (Sect. 5.1), then refines the corner positions of each marker (Sect. 5.2), and finally uses the resulting data to reconstruct the robot 3D position in the W reference frame (Sect. 5.3).

This techniques basically reproduces the functionality of ARToolkit, but provides a significantly higher accuracy at the expense of an increased computation time. Section 7 details the comparison between GTvision using the developed system, and GTvision using the original ARToolkit.

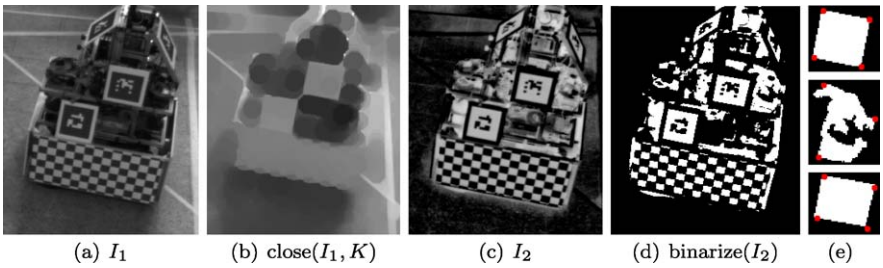


Fig. 7 (Color online) (**a**–**d**) Image processing steps for marker detection (see text); (**e**) filled masks of candidates passing all initial tests, with detected corners (*red*); the middle candidate will be discarded later in the process

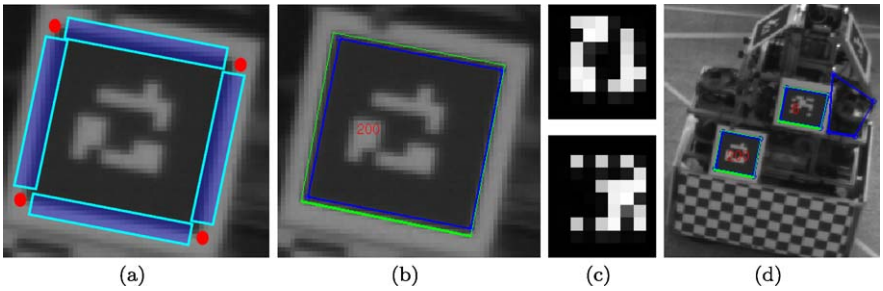


Fig. 8 (Color online) (**a**) Rough corners from previous steps (*red*), and areas where border lines are fit (*blue*). (**b**) Rough marker edges (*blue*) vs. refined marker edges (*yellow*): the maximum distance is about 3 pixels, which would reflect to significant reconstruction errors. (**c**) Recovered ID matrices. (**d**) Resulting image, with markers decoded; note false detection (*rightmost blue quadrangle*), rejected as not validated in the refinement stage

5.1 Detection of visual markers

The marker detection step analyzes every frame of the input video and detects visible markers, providing rough estimates of the coordinates of the corners of each visible marker.

First, a set C of candidate markers in the given frame is built and a roughly-segmented binary mask M_i representing each marker is provided. This phase is designed to be very sensitive, in order to reduce the probability of missing valid candidates; hence a significant amount of invalid candidates (i.e., candidates not being actual markers) are likely to be detected in this phase and returned in set C; they are discarded and filtered in the following.

Elements in C are added as their brightness pattern is compatible with the presence of a marker. In particular, the original image I_1 is divided by its own grayscale closure, then inverted:

$$I_2 = 1 - \frac{I_1}{\text{close}(I_1, K)}, \tag{1}$$

where K is a disk-shaped kernel with a radius roughly as large as the expected image of the marker. The effect of the closure operation is to fill the area with the white grey-level surrounding the marker. As a consequence, in I_2 the square markers have large intensity values; I_2 is then binarized, and each of its connected components is considered a candidate marker with binary mask M_i (see Fig. 7).

All elements of C are now subject to a number of morphological tests on their binary masks M_i in order to discard obvious misidentifications. In particular, all of the following conditions must be met by each marker mask:

Size, which should be compatible to the expected area of a marker; although we tolerate a wide variation in the measured marker area vs. the expected area (up to 6 times bigger or smaller), this test still usually discards most of the candidates in a crowded scene.

Euler number: M_i is expected to have at least one hole, caused by the white cells in the marker ID matrix; this corresponds to enforcing that the Euler number of M_i is at most 0.

Fraction of the marker image covered by holes: as roughly half of the cells of the marker ID matrix are white, and the ID matrix itself covers $1/4$ of the marker area, we expect holes in M_i to cover about $1/8$ of its total area.

Solidity, i.e., the fraction of the marker's convex hull which overlaps with the marker itself; as the marker projects to a convex quadrangle, we expect this to be close to 1.

We then implement a final test on the surviving candidates by looking for four well-defined corners on the filled marker mask; when our routine detects such corners, the candidate is accepted and passed to the subsequent step, where such approximate corner localization will initialize the refinement of the marker localization in the image. Localizing well-defined corners from the binary marker image is not trivial, because most simple algorithms are fooled by small errors due to imprecise binarization. We use an ad-hoc iterative algorithm, which searches and refines a set of four

roughly evenly-spaced points on the chain of perimeter pixels of M_i, such as those where the quadrangle they define has the largest area.

5.2 Refinement of marker corners and ID decoding

Once the rough corner points of a marker i are known, the binary mask M_i used for the previous processing is disregarded, and the original image data is again taken into consideration in order to refine as much as possible the localization of the four corners; this is especially critical as the subsequent 3D reconstruction step heavily depends on the accuracy of this corner localization.

In order to provide accurate results, we localize the corners of each marker by considering *all* the pixels along its four external edges. This contrasts with faster but less accurate techniques such as the ARToolKit, which instead use only local information for corner localization. Such techniques are particularly sensitive to noise and systematic errors: in particular, due to defocus and motion blur, corners often appear smoothed and are misplaced even by one or more pixels towards the center of the marker. By exploiting a much larger amount of data, our technique reaches superior accuracy, noise resiliency and robustness.

We first precisely localize the four straight lines surrounding the marker by using data from all the pixels near the expected position of such edges (except those near the corners, which are deemed unreliable as previously introduced)—see Fig. 8. The expected position of the edges is computed from the rough localization of the corners provided by the previous phases. The four lines are then (algebraically) intersected in order to recover the refined corner points. Any error in this process leads to a late rejection of the candidate marker. One such possibility is the failure to find a well-fitting line to the edge in the image area where the marker border is expected (which we have found to be a rare occurrence).

Once the refined corner points are known, they are exploited in order to rectify the marker image and recover the binary matrix represented inside. If that does not match with any known marker, the marker is rejected.

5.3 Reconstruction of the robot pose

For every input frame, the set of the detected markers is used in order to recover the 3D position of the robot. As a first step we compute the 3D position of each corner of each detected marker in the reference frame of the robot, by exploiting the known rigid transformation between the robot frame and the marker itself. Then we exploit the known location of the image points associated to each marker corner, which we have precisely localized in the previous steps.

In short, if r markers are detected in a frame, we have $4 \cdot r$ image points (from a single calibrated camera), paired with $4 \cdot r$ 3D points in the frame of reference of the robot. Since the camera is calibrated, each of the image points can be backprojected to a viewing ray in world coordinates. Our goal is to recover the 3D rigid transformation leading from the world reference frame to the robot reference frame, such that the 3D points project to the known image points. This corresponds to the well-known PNP problem (Perspective N-Point—in our case $N = 4 \cdot r$): such problem is not trivial, but several different solution techniques are available, e.g., Schweighofer and Pinz (2008), Moreno-Noguer et al. (2007) and Lu (1999). In particular, we use the algorithm proposed in Schweighofer and Pinz (2008), which we have found to be the most robust in this case, especially when only one marker is visible—a frequent occurrence in our data.

6 Ground truth from laser scanners

GTlaser data is derived by means of a number of fixed laser scanners placed in the environment. As described in Sect. 4.3, their relative positions are known and referred to the world reference frame W. The following procedure is adopted in order to analyze scans acquired during robot motion, for reconstructing its pose. Readings from the four laser scanners are initially aligned to the common reference frame W; this is made possible by the calibration procedure described in Sect. 4.3. Then, we use the popular Iterative Closest Point (ICP) algorithm to recover the pose of the robot in the environment.

In order to provide a robust system, we implemented a filtering technique to remove noise from the data gathered by the laser scanners, to remove points not belonging to the robot outline. In particular, we implemented an approach similar to the classical background subtraction technique used in video-surveillance:

- laser scans obtained from the static environment are used to build a "background scan" of it;
- the points generated by the environment (such as walls) are removed from the scans by excluding readings that are close in space to the background scan;
- points which are not close to the last acquired profile of the robot are also discarded as noise too.

At the end of this processing we obtain a scan where only a very reduced amount of noise is present; the ICP algorithm (Besl and McKay 1992) is used to align two point sets by computing the rigid transformation between them. In our case such two sets are defined as:

- the cloud of points obtained by fusing the laser scanner data and deleting all background and noise as mentioned before; these scan points are in the world frame;

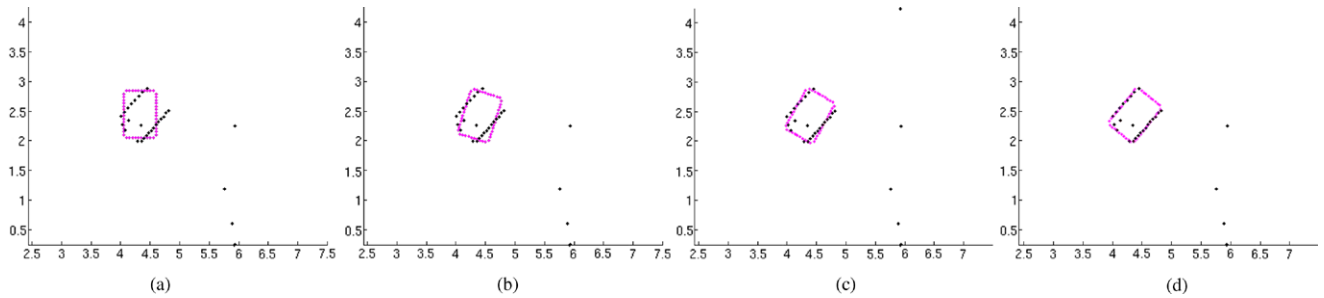


Fig. 9 (Color online) An example of ICP execution where the pose initialization has been intentionally set in a suboptimal way. Filtered laser data (*black*) and robot model (*magenta*) at some of the ICP iterations

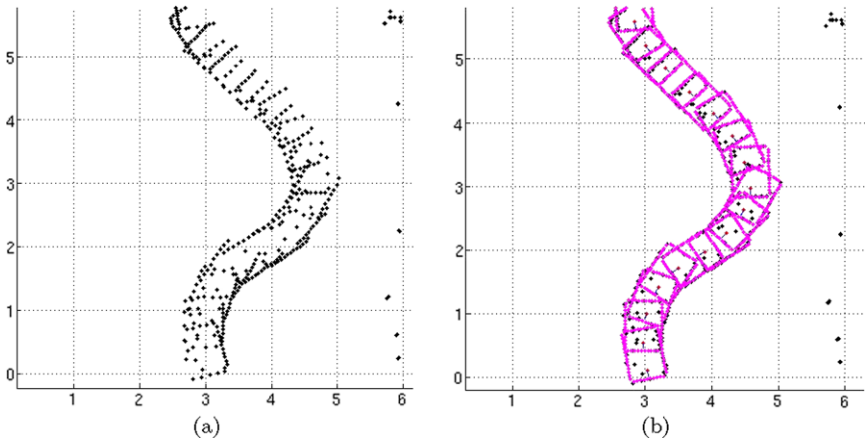


Fig. 10 (**a**) Laser scans after filtering, overlapped from several consecutive acquisitions; (**b**) the recovered robot outline for each acquisition

– the shape model of the robot, represented by a set of points along the rectangular contour of the robot, which is expressed in the reference frame of the robot.

The resulting rigid transformation represents the robot pose. In Fig. 9 a few iteration of the GTlaser approach is presented where the algorithm has been intentionally initialized with a quite suboptimal rigid transformation between the two sets of points. In the real execution, we initialize the orientation with the previous known position of the robot, in order to ease and speed-up convergence: as the scanners run at 75 Hz the previous pose is usually very near to the current one; the results on a chunk of data is reported in Fig. 10.

7 Experimental results and validation

We worked with two different indoor setups, in different places. Both environments were equipped with calibrated cameras and laser scanners. In particular, in the first experiment, performed in a room called GTroom in the sequel, GTvision has been computed, along with a set of manually acquired validation GT measurements ("*GTvalidation*"). The experiment in the GTroom allowed us to compare the original ARToolkit-based approach with the more accurate algorithm that we are proposing to use.

In the latter experiment, which was performed in the Bicocca location, i.e., the specific location where the RAWSEEDS indoor acquisition campaigns have also been performed, we computed both GTvision and GTlaser, and we could measure the metric errors of both, again with respect to GTvalidation, so that a comparison could be performed. Differently from what happened in the GRroom, the GTvalidation measurements could not be gathered manually because of the lower accuracy implicated by the much larger size of this location. To tackle this problem, i.e., to get accurate GTvalidation measurements, we used the robot onboard laser scanners to improve manual measurements.

7.1 Acquisition of validation ground truth in the GTroom

In the GTroom we manually measured the pose of the robot in 26 different positions along its trajectory. These measurements have been used for the validation of the results of the GTvision collection methods.

In order to manually measure the robot pose, we accurately measured the positions of 16 points in the room (world points) in the world reference frame. Moreover, we also defined relevant points on the robot, whose coordinates are known with respect to the robot itself: we selected the extrema of the robot-frame axes. We manually moved the robot around the room, to the poses where we gathered the

validation measurements. Then, for each such robot pose, we collected the validation data. This meant to project, i.e., to draw, on the floor the robot-points and then, for each robot-point, to measure their distance from 16 known world points. These measurements were performed with a laser meter whose usage is typical in civil engineering, having a range of several tens of meters with an accuracy of some millimeters. These data allowed us to compute the pose of the robot frame, along with its uncertainty, for each of the 26 validation poses.

Table 1 Details of the metric error for both GTvision methods with respect to GTvalidation. Note that ARToolkit detects less markers than our detector

	GTvision	
	Our detector	ARToolkit
Average error [cm]	9.53	17.02
Maximum error [cm]	14.53	36.70
Standard deviation [cm]	2.35	8.72
Number of poses localized	25/26	15/26

The robot pose estimates were then compared to the output of the GT systems, in order to evaluate their accuracy. We checked whether the validation system was accurate enough by reconstructing the position of a point 20 mm from a known one. The outcome was accurate to less than 5 mm.

7.2 Preliminary GTvision validation and comparison with ARToolkit

The validation setup in the GTroom has been used to compare the accuracy of GTvision with the GTvalidation data,

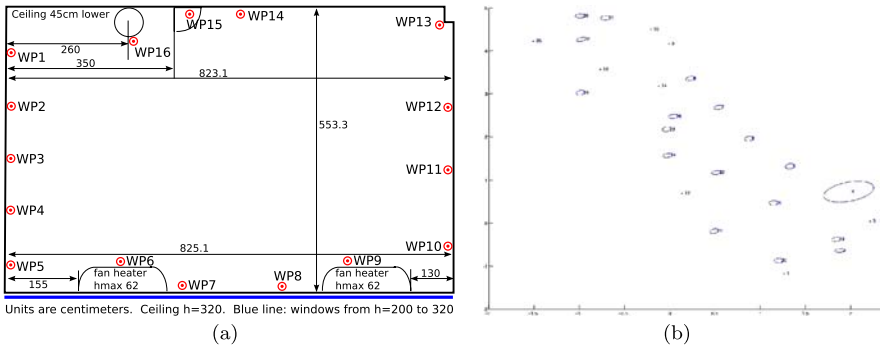


Fig. 11 (Color online) (**a**) Blueprint of the GTroom, with the 16 world points marked as *red circles*; (**b**) the 26 recovered robot poses included in GTvalidation, with the computed covariance ellipses

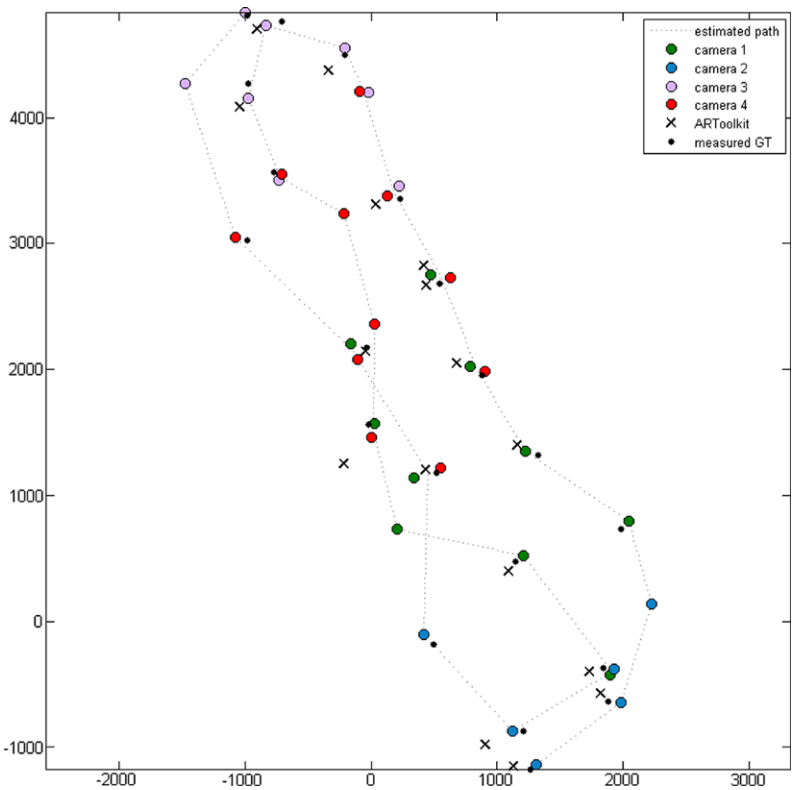


Fig. 12 Performance comparison of GTvision, GTvision (ARToolkit), and GT Validation

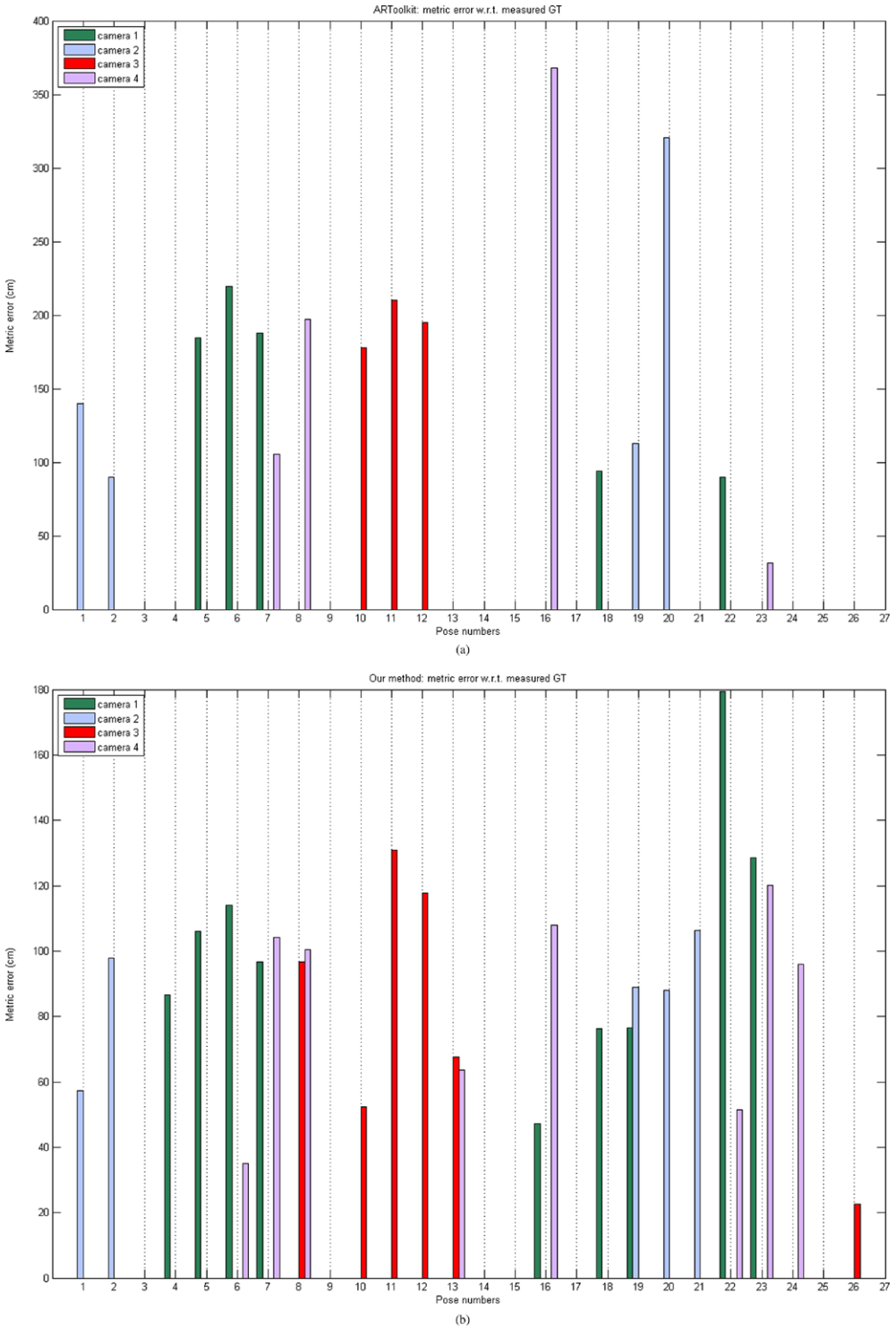


Fig. 13 Details of the metric errors of both GTvision methods with respect to the GTvalidation. Note that ARToolkit detects much less markers than our detector

in the 26 poses where the latter is available. More interesting and unique to the GTroom experiment (the comparison with the validation poses has been also performed in the actual location where the datasets have been gathered), is the comparison of the results of our GTvision technique with the results obtained by localizing markers using ARToolkit.

Table 1 shows the results of GTvision when using ARToolkit, and when using our improved detector: notice that our detector is more accurate in estimating the poses and

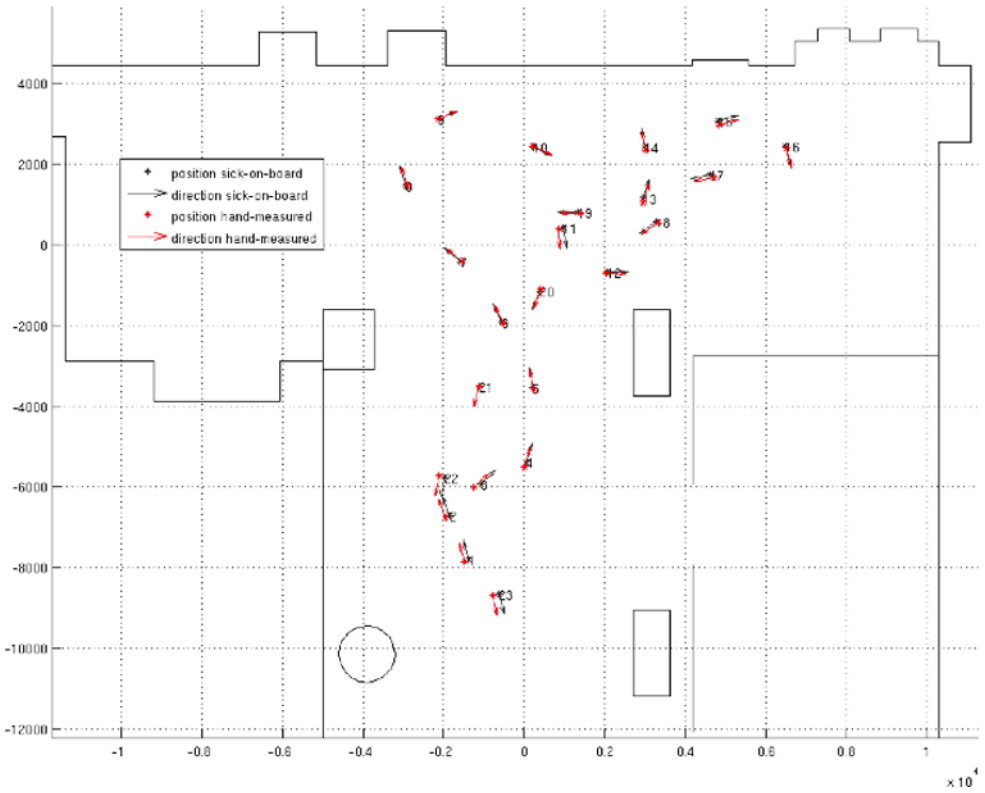


Fig. 14 Comparison of the validation poses obtained from the manual procedure and the ICP-based one

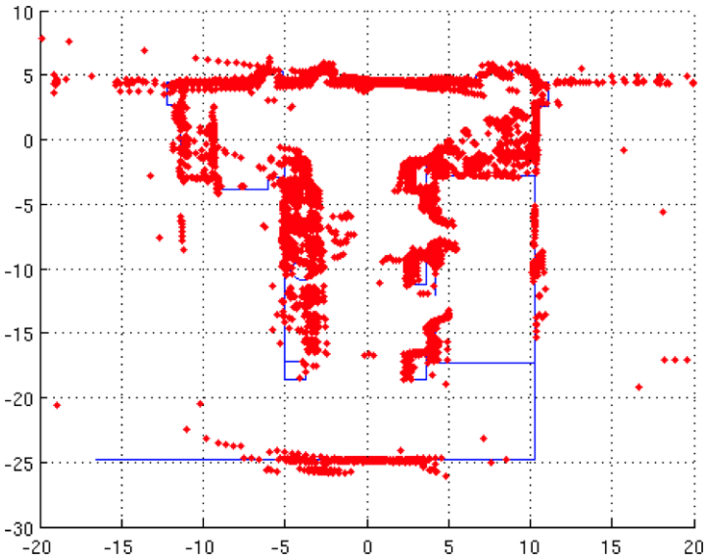


Fig. 15 Onboard scans in the validation poses, using the manually determined values for the poses. Cluttered areas correspond to tables and chairs ($x = -4$ m, $y = 0 \ldots -15$ m) and to the area where RAWSEEDS' equipment was installed ($x = 5 \ldots 10$ m, $y = -2$ m)

manages to localize the robot in more frames. Figure 12 shows the location of these poses along the robot trajectory and the poses estimated with the different methods. In Fig. 13 we show the detailed metric error of each pose for both methods: notice that the resulting mean metric error of the GTvision is less than 100 mm, which represents an acceptable results for our needs.

7.3 GTvision and GTlaser validation in the Bicocca location

In Bicocca location we could not base only on the manual procedure used in the GTroom, for gathering validation robot poses. The reason is the reduced accuracy of this triangulation-based procedure in the much larger GT area defined in the Bicocca location, see Fig. 6. In order to get

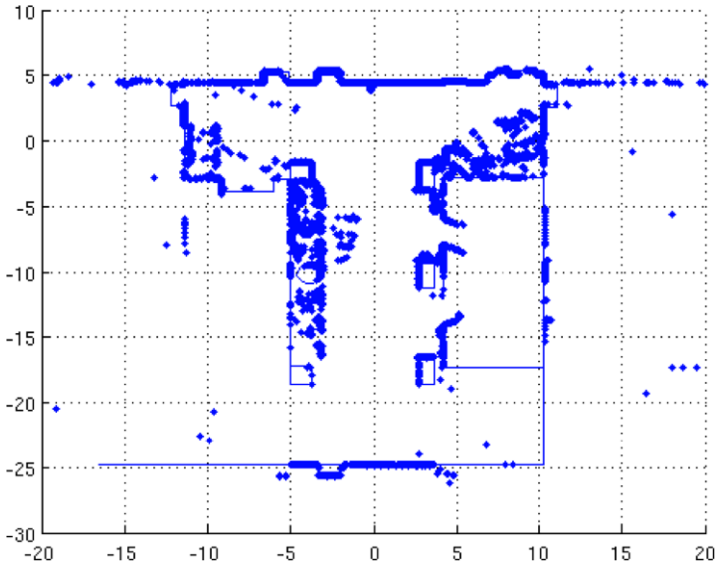


Fig. 16 Onboard scans in the validation poses, using the ICP-based values for the poses

estimates as accurate as in the GTroom we had to exploit the onboard laser scanner. While performing this activity we also discovered that the map, as in the available blueprints, included some mistakes, which we manually corrected, see Fontana et al. (2009) for details.

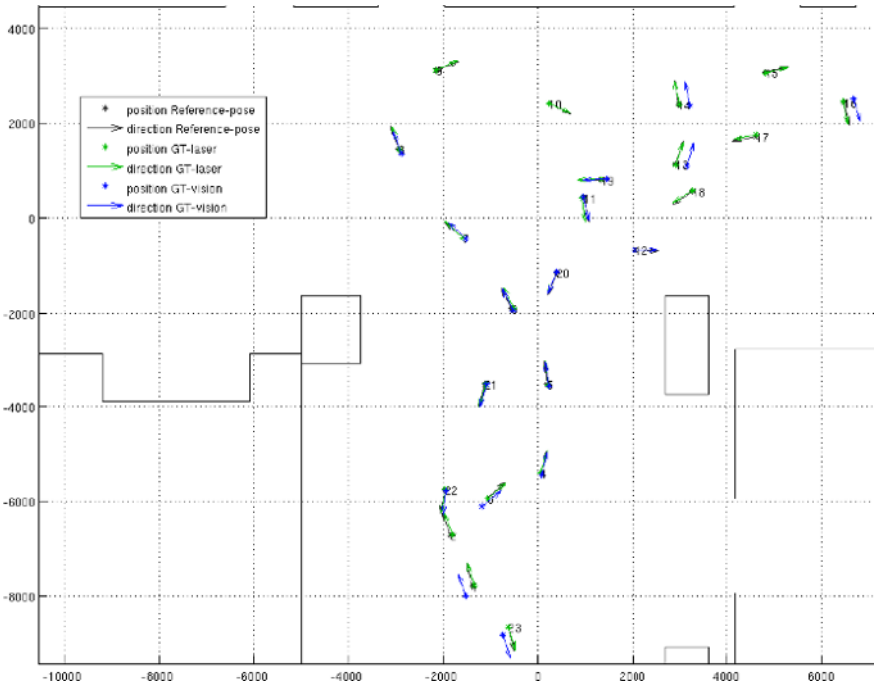
The determination of the validation poses exploits ICP, in order to align the onboard scans of each pose to the world map. This is obtained as follows: the (less accurate) pose obtained with the manual procedure followed in the GTroom is used as initialization for an ICP-based alignment of the two onboard scans to the world map. In the Bicocca location we defined 20 world points and 23 validation poses, see again Fontana et al. (2009) for details. The 23 validation poses can be observed together in Fig. 14, though just by looking at this figure we cannot say which system is the more accurate. In order to show that the quality of the manual procedure is not accurate enough in the larger GTarea of the Bicocca location, we drew all the scans gathered in the validation by using the manually-determined pose, together with the map, see Fig. 15. This can be compared with the similar figure obtained by drawing the onboard scans from the ICP-based poses, see Fig. 16. The latter figure is also the only way we had to check the absolute quality of the values of the validation poses, and it cannot be possible to give a precise estimate of the error for this improved GTvalidation, in comparison to the manual one.

Exploiting the ICP-aligned poses, we could determine the accuracy of the GTpose collection systems that we set up. Figure 17 shows the output of the two systems for the 23 validation poses. From Fig. 18 it is also easy to notice that the poses reconstructed by GTlaser are more precise than those from GTvision, particularly when we consider poses that are far from the world reference frame. This is coherent with the inner working of the GTvision system, as in such cases the robot is observed by cameras having a reference frame that is linked to the world reference frame by a chain of multiple roto-translations. Errors for GTlaser are always smaller than 50 mm, while orientation are typically within a range of ± 2.5 degrees. GTvision errors are characterized by large peaks together with much more accurate results, for some poses even better than GTlaser. For GTlaser the mean linear distance error amounts to 20 mm, with a standard deviation of 11 mm; the mean angular distance error amounts to 0.15 degrees with a standard deviation of 1.56 degrees. For GTvision the mean linear distance error amounts to 112 mm, with a standard deviation of 90 mm; the mean angular distance error amounts to -0.80 degrees with a standard deviation of 2.16 degrees. It is worth noticing that in several poses the angular errors are of opposite sign, which is expected because of the independence of the two systems. Therefore a proper combination of GTvision and GTlaser could compensate them obtaining an even better estimate of robot pose.

Figure 19 shows the results of the online execution. In this case we localized the pose of the robot in each frame of the camera videos. The GTvision and GTlaser data acquired were spatially aligned with the world reference frame W and aligned in time by matching the timestamps. The relative distances of corresponding poses for the two data stream were calculated: the average metric error was 93.5 mm with a standard deviation of 42.0 mm, which are again acceptable for our needs.

7.4 Integration of visual markers, laser, and odometry data

Although the GTlaser and GTvision data are theoretically independent and could be used by themselves, they can be fused with the readings from the robot's odometry sensor, in order to obtain a more reliable measurement of the robot's



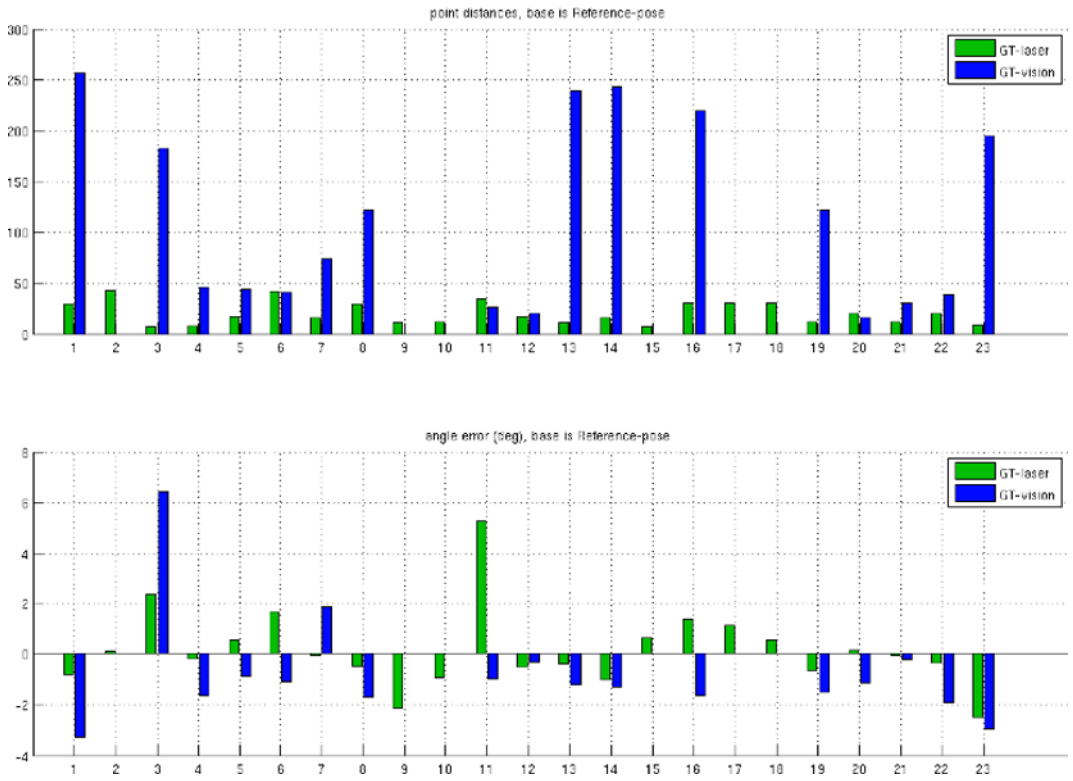


Fig. 17 (Color online) The 23 validation poses in *black*, the 23 poses reconstructed by GTlaser in *green*, the 23 poses reconstructed by GTvision in *blue*. Please note that for poses 9, 5, 15, 17, 18 GTvision was unable to localize the robot. Measurement units are millimeters

Fig. 18 (Color online) Differences between the output of GTlaser and GTvision. In particular, linear and angular errors with respect to the validation poses. GTlaser in *green*, GTvision in *blue*; measurement units are millimeters and degrees

real position in the environment. To do this, we can assume the uncertainty associated with the source measurements to be Gaussian: therefore, we can use a classical Kalman approach to perform data fusion.

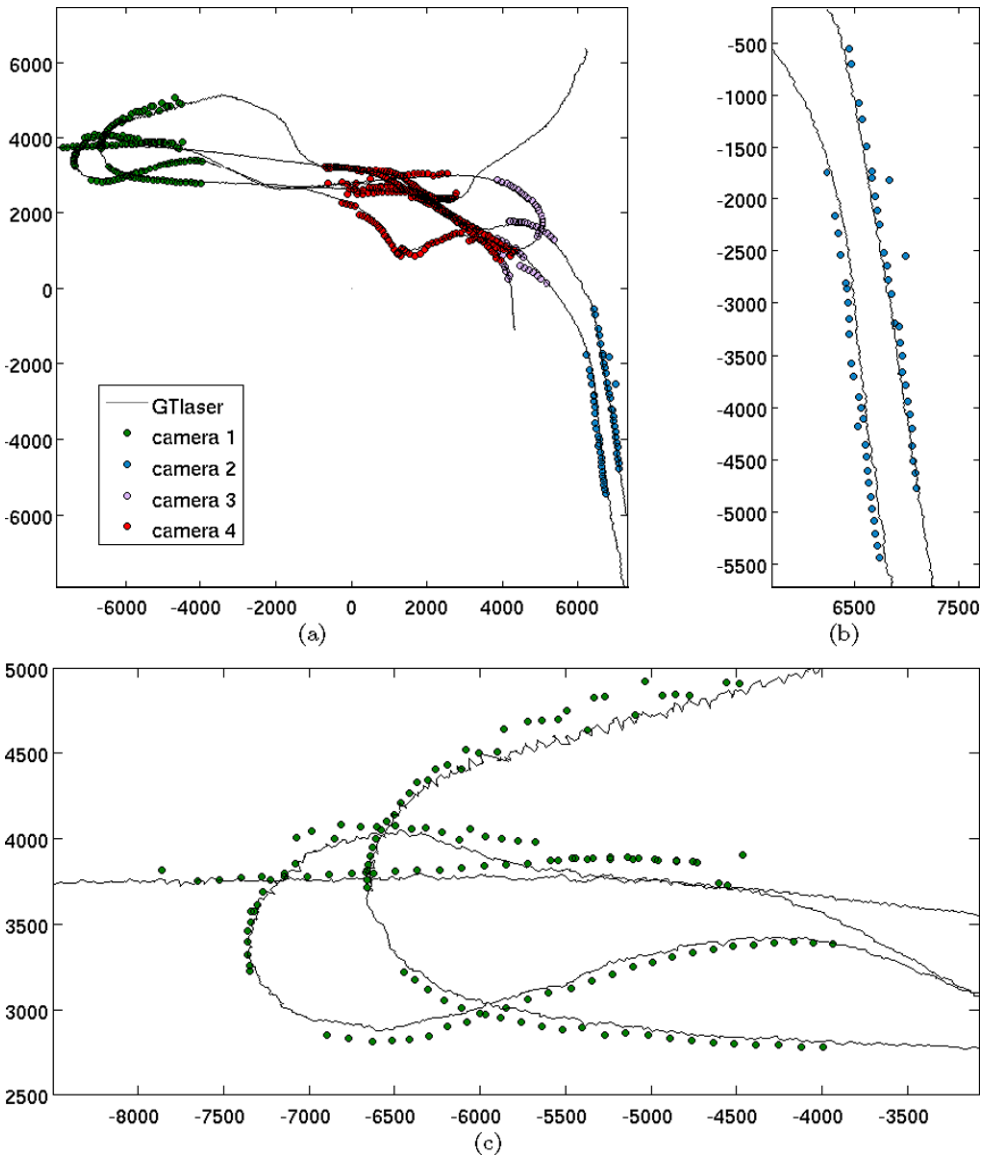


Fig. 19 (**a**) Comparison of GTlaser and GTvision for the online execution. (**b**, **c**) Close up of the area covered by two cameras of the network

In particular, when both streams are available, we use an Extended Kalman Smoother, which beside optimally integrating the two independent estimates according to their accuracy, also allows to further improve the estimate of the robot position by using the both past and future information, at each time stamp. This is applicable in our case as data fusion can be performed as a postprocessing step, when the whole dataset is available.

The algorithm is composed by two phases: a forward filtering (just like the classical Extended Kalman Filter), followed by a backward recursion that smooths the past estimates, integrating the future ones. Being a detailed description of the Kalman Smoother out of the scope of this paper, we point the interested reader to Yu et al. (2004); here we just report the fact that, by fusing GTlaser and GTvision according to their uncertainties, we can obtain an accuracy of 5 cm and 0.5 degrees.

8 Discussion and conclusions

In this paper we dealt with the often overlooked problem of collecting ground truth data, which is a critical step for benchmarking self-localization and SLAM algorithms. We described two independent techniques, GTvision and GTlaser, which do not rely on the robot's own sensor suite, but instead use external cameras and external laser scanners, respectively; we compared and validated their performance also using manually-acquired validation data, whose uncertainty is very low, at the expense of a major effort required to acquire them.

Results show that both techniques provide sufficient accuracy for most uses; in particular, we have shown that the technique exploiting visual markers needs to locate them very accurately in the images, and required a more accurate approach in comparison to off-the-shelf systems such as the ARToolkit (which is designed for efficiency rather than accuracy). We have also underlined the many issues related to ground truth data gathering using these techniques—namely calibration of cameras, estimation of rigid transformations between markers, and alignment of laser sensors—along with our solutions, whose validity has been proved by experimental validation.

Each of the two proposed GT-collection systems has its own pros and cons so the choice of which one should use has to take both of them into account. GTlaser can be employed in order to obtain more precise measurement than GTvision (the laser range finders used in our experiments provided errors in the range of a few centimeters) and it can get this measurement online. Moreover each single sensor has a wider horizontal field of view than a perspective camera and thus allows one to use less sensors to cover the same area. However, using linear laser range finders the system can reconstruct the robot position in three degrees of freedom, and thus it is not suited for non planar movement; in addition to this, such a system requires a well defined robot profile (i.e., the gown).

GTvision is undoubtedly more cumbersome in setup and the present implementation is meant for off-line computation, but it allows the computation of a six degree of freedom pose for the robot. GTvision does not require the presence of the gown, but exploits a set of markers. Moreover, it has to be noticed that illumination has to be reasonably uniform in order to obtain a good localization. From a cost perspective, the hardware cost for this system is much lower than the GTlaser one, but we have to consider that setup and calibration activities bring their own costs.

When it is feasible, both GT-collection systems could be used and their estimates combined as suggested in Sect. 7.4, but in case only one of them has to be chosen this choice has to be made by taking into account the algorithm to be benchmarked. In case we are interested in a full six degrees of freedom SLAM algorithm to be benchmarked the choice is forced toward the GTvision method, while in case we are interested only in three degrees of freedom the obtainable accuracy suggests to go for the GTlaser solution. However, this consideration does not take into account the environment in which the ground truth is collected; in very cluttered environments such as offices or libraries the presence of several objects on the floor (e.g., tables, chairs, bags, trolleys, etc.) make the field of view of the laser occluded so the camera solution has to be preferred instead. Moreover, the GTlaser is perfectly suited for wide halls while in long corridors the number of points reflected by the gown might be small and grouped on just one of the gown sides; in this situation there might be not enough information to obtain a good estimate both in position and orientation.

An orthogonal issue might be related to the appropriateness of the accuracy obtained with the ground truth systems presented in this paper when measuring the results of today's SLAM algorithms. This is not an easy answer to give since, nowadays, a clear quantification of the accuracy in SLAM independently from the sensor used is not possible, and indeed this is the final aim of the RAWSEEDS project. Some kind of feeling about the "accuracy bound" obtainable from a SLAM algorithms can be obtained by looking at the accuracy of the sensor used. Some concerns might arise when SLAM algorithm using laser scanners are compared using our ground truth since the accuracy of a laser range finder is definitely higher. Conversely, when we use vision sensors to perform SLAM, the uncertainty in the measurements, especially when features are far away from the observer, is higher than the accuracy obtained by our systems.

References

Angle, C. (2004). Inv. talk at euron meeting. The author is CEO of iRobot inc.

Besl, P. J., & McKay, N. D. (1992). A method for registration of 3-d shapes. *IEEE Transactions on PAMI*, *14*, 239–256.

Bouguet, J. (2002). *Camera calibration toolbox*. Available at http://www.vision.caltech.edu/bouguetj/calib_doc.

Durrant-Whyte, H. F. (1987). *Integration, coordination and control of multi-sensor robot systems*. Norwell: Kluwer Academic.

Folkesson, J., & Christensen, H. I. (2004). Robust slam. In *5th IFAC symp. on intelligent autonomous vehicles*.

Fontana, G., Marzorati, D., Giusti, A., Taddei, P., & Rizzi, D. (2009). *Deliverable 2.1—part 2: Ground truth collection and validation*. Project RAWSEEDS deliverable, available at www.rawseeds.org.

Howard, A., & Roy, N. (2003). *The robotics data set repository (radish)*. http://radish.sourceforge.net.

Kato, H., & Billinghurst, M. (1999). Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *IWAR '99: proceedings of the 2nd IEEE and ACM international workshop on augmented reality* (p. 85). Washington: IEEE Comput. Soc.

Lu, C. P. (1999). Fast and globally convergent pose estimation from video images. *Transaction on Pattern Analysis and Machine Intelligence*.

Lu, F., & Milios, E. (1997). Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, *4*, 333–349.

Moreno-Noguer, F., Lepetit, V., & Fua, P. (2007). Accurate non-iterative o(n) solution to the pnp problem. In *Proceedings of ICCV*.

Pirjanian, P. (2007). Panelist at the ICRA 2007 panel future of robot operating systems. Author is with Evolution Robotics; www.evolution.com.

Schweighofer, G., & Pinz, A. (2008). Global optimal o(n) solution to the pnp problem for general camera models. In *Proceedings of BMVC*.

Sorrenti, D. G., & Matteucci, M. (2008). *Deliverable AD2.3: Validation of the ground truth collection systems*. Project RAWSEEDS deliverable, available at www.rawseeds.org.

Tardós, J. D., & Castellanos, J. A. (1999). *Mobile robot localization and map building: a multisensor fusion approach*. Norwell: Kluwer Academic.

Thrun, S., Montemerlo, M., Koller, D., Wegbreit, B., Nieto, J., & Nebot, E. (2004). Fastslam: An efficient solution to the simultaneous localization and mapping problem with unknown data association. *Journal of Machine Learning Research*.

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

Ubisense, O. (2006). *Ubisense hardware datasheet*. http://www.inition.co.uk/inition/pdf/mocap_ubisense_hardware.pdf.

Yu, B. M., Shenoy, K. V., & Sahani, M. (2004). *Derivation of extended Kalman filtering and smoothing equations*.

Simone Ceriani was born in 1983 in Tradate, Italy. He got first level laure degree in 2005 and master level degree in 2008 in Computer Science Engeenering. His principal interests are mobile robots and computer vision. Since 2007 Simone has been part of the AIRLab team, in particular involved in the LURCH project, whose aim is to develop an autonomous wheelchair to help disabled people. He is currently enrolled as Ph.D. Student at Politecnico di Milano.

Giulio Fontana received the "Laurea cum laude" degree in electronic engineering from Politecnico di Milano in 2002. Since then, he has worked as a research assistant at the Artificial Intelligence and Robotics Laboratory of the Politecnico di Milano (http://www.airlab.elet.polimi.it/). His work is mainly oriented towards the design and implementation of advanced robotic systems and architectures; however, he tries to keep a global viewpoint on research and technology—especially where hardware is involved—in the fields of robotics, AI and computer vision. He participated in several national and international research projects, and presently he is WorkPackage Leader in project RAWSEEDS (Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets, funded by the European Commission (http://www.rawseeds.org).

Alessandro Giusti received his B.S., M.S. and Ph.D. degrees in Computer Engineering from Politecnico di Milano, Italy in 2003, 2005 and 2009, respectively. In 2008 he worked as an intern at the Joint Research Center of the European Commission. His main research interests are in Computer Vision and Image Analysis, in particular related to motion blur theory and interpretation, 3D vision, and biomedical image analysis applied to embriology. He is currently a researcher at Dalle Molle Institute for Artificial Intelligence (IDSIA) in Lugano, Switzerland.

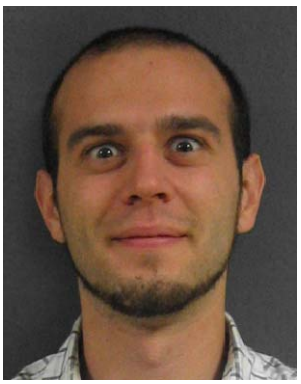
Daniele Marzorati was born in Milan, Italy, in 1980. He got his Laurea (Master) Degree in Computer Science at Università degli Studi of Milano—Bicocca in April 2005. He received a Ph.D. degree in Computer Science from the Department of Computer Science at the Università degli Studi of Milano—Bicocca in February 2009. His main research interests are robotics and machine perception systems (in particular artificial vision systems). He is currently a PostDoc at the Università degli Studi of Milano—Bicocca.

Matteo Matteucci ("Laurea" degree 1999, M.S. 2002, Ph.D. 2003) is Assistant Professor at Politecnico di Milano. Divided between Robotics and Algorithms for Machine Learning, in 2002 he received a Master of Science in Knowledge Discovery and Data Mining at Carnegie Mellon University (Pittsburgh, PA), and in 2003 a Ph.D. in Computer Engineering and Automation at Politecnico di Milano. He is actually working in both Robotics and Machine Learning, mainly applying, in a practical way, techniques for adaptation and learning to autonomous robotics systems. He is part of the Technical Committee of AI and Robotics conferences, and reviewer for international journals on Robotics and Computational Intelligence. He is the coordinator of the RAWSEEDS project (2006–2009) an FP6 Specific Support Action to develop a benchmark toolkit for SLAM.

Davide Migliore was born in Milan, Italy, in 1981. He received the Laurea degree in Computer Science Engineering in 2005 from the Politecnico di Milano with the final mark of 100/100 cum laudae. He received a Ph.D. in Computer Engineering at Dipartimento di Elettronica e Informazione at Politecnico di Milano, Milano, Italy in April 2009 with a scholarship funded by Italian Institute of Technology (IIT) on "Robotics for disabled people". Currently the he is a Postdoc at AIRlab on the project "Navigation algorithm for an autonomous wheelchair". His research is in navigation systems for mobile robots, Simultaneous Localization and Mapping (SLAM), robotics for impaired people, algorithms for videosurveillance and environment monitoring, objects recognition and uncertain geometry.

Davide Rizzi received a B.S. degree from Politecnico di Milano, in 2007. Since then he actively collaborated to the RAWSEEDS project, as a software engineer and developer, hardware maintainer and data gathering, processing and formatting man. His principal interests include mobile robotics, sensors, embedded systems, and hardware/software integration.

Domenico G. Sorrenti got his Maturità Classica in 1981 from Liceo G. Carducci Milano, his Laurea (Electronic Engineering) from Politecnico di Milano in 1989, and his Ph.D. (Computer and Control Engineering) also from Politecnico di Milano in 1992. After a period with "Vision and Robotics Lab.", King's College London, in 1995 he began as ricercatore (assistant professor) with the Scienze dell'Informazione Dept. of Università degli Studi di Milano. In 1999 he moved to the Informatica, Sistemistica e Comunicazione Dept. of Università degli Studi di Milano—Bicocca. Since 2005, he has been an associate professor with the same department.



Pierluigi Taddei received a Master Degree in Computer Science from the University of Illinois at Chicago and a Master cum laude in Computer Engineering from Politecnico di Milano in 2005. He then received a Ph.D. in Computer Vision from the same department in April 2009. He collaborated with multimedia companies as software engineer on user interfaces and as computer graphic 3d modeller. He is currently a postdoc researcher at the IPSC unit of the European Joint Research Center. His research interests involves 3D reconstruction of deformable surfaces, visual odometry using panoramic and non central cameras and 3d scene change detection exploiting laser scanners.