

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/220474374>

A collection of outdoor robotic datasets with centimeter-accuracy ground truth

Article in *Autonomous Robots* · November 2009

DOI: 10.1007/s10514-009-9138-7 · Source: DBLP

CITATIONS

148

READS

618

3 authors:



Jose Luis Blanco

Universidad de Almería

147 PUBLICATIONS 2,517 CITATIONS

[SEE PROFILE](#)



Francisco Angel Moreno

University of Malaga

27 PUBLICATIONS 618 CITATIONS

[SEE PROFILE](#)



Javier González-Jiménez

University of Malaga

327 PUBLICATIONS 6,071 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Incorporation of contextual exploitation to convolution neural networks applied to mobile robotics. [View project](#)



PROMOVE: Advances in mobile robotics for promoting independent life of elders [View project](#)

A Collection of Outdoor Robotic Datasets with centimeter-accuracy Ground Truth

Jose-Luis Blanco · Francisco-Angel Moreno · Javier Gonzalez

Received: date / Accepted: date

Abstract The lack of publicly accessible datasets with a reliable ground truth has prevented in the past a fair and coherent comparison of different methods proposed in the mobile robot Simultaneous Localization and Mapping (SLAM) literature. Providing such a ground truth becomes specially challenging in the case of visual SLAM, where the world model is 3-dimensional and the robot path is 6-dimensional. This work addresses both the practical and theoretical issues found while building a collection of six outdoor datasets. It is discussed how to estimate the 6-d vehicle path from readings of a set of three Real Time Kinematics (RTK) GPS receivers, as well as the associated uncertainty bounds that can be employed to evaluate the performance of SLAM methods. The vehicle was also equipped with several laser scanners, from which reference point clouds are built as a testbed for other algorithms such as segmentation

or surface fitting. All the datasets, calibration information and associated software tools are available for download¹.

Keywords Dataset · Least squares · GPS localization · Ground truth · SLAM

1 Introduction

1.1 Motivation

The field of Simultaneous Localization and Mapping (SLAM) has witnessed a huge activity in the last decade due to the central role of this problem in any practical application of robotics to the real life. While the theoretical bases of SLAM are well understood and are quite independent of the kind of sensors employed by the robot, in practice many of the reported works focus on either 2-d SLAM (assumption of a planar world) or 6-d SLAM (features have full 3-d positions and the robot can also rotate freely in 3-d). Typically the first group of works rely on laser scanners while the latter employ monocular or stereo cameras.

A critical issue which must be considered for any SLAM techniques, either 2-d or 6-d, is the *evaluation of its performance*. Usually, more recent techniques claim to be *better* in any sense with respect to previous works. Here, *better* can mean more accurate (in the case of building metric maps), less prone to divergence or more scalable, among other possibilities.

In principle, the advantages of some techniques in comparison to others should be *quantified*, but that is not a goal easy to achieve in practice. In some cases, the differences between two methods are more qualitative

This work has been partly supported by the Spanish Government under research contract DPI2005-01391 and the FPU grant program.

Jose-Luis Blanco (Corresponding author)
E.T.S.I. Informática, Lab. 2.3.6
University of Málaga
29071 Málaga, Spain
E-mail: jlblanco@ctima.uma.es

Francisco-Angel Moreno
E.T.S.I. Informática, Lab. 2.3.6
University of Málaga
29071 Málaga, Spain
E-mail: famoreno@isa.uma.es

Javier Gonzalez
E.T.S.I. Informática, Office 2.2.30
University of Málaga
29071 Málaga, Spain
E-mail: jgonzalez@ctima.uma.es

¹ <http://babel.isa.uma.es/mrpt/papers/dataset2009/>

than quantitative, but most often measuring the accuracy of the results becomes necessary. Instead of contrasting the maps produced by the different methods (which usually relies on visual inspection), it is more convenient to consider the robot *reconstructed paths* due to its reduced dimensionality in comparison to the maps. Additionally, the evaluation of robot paths would even enable comparing a 2-d method (based, for instance, in grid mapping) to a 6-d technique such as vision-based SLAM. This would not be possible if the maps, of different nature, were instead employed in the comparison.

The traditional problem found by the SLAM community in this sense is the lack of a reliable *ground truth* for the robot paths. Some works have to rely on simulations to overcome this difficulty, but this approach ignores the problems that arise with real-world sensors. The existence of public reference datasets with an accurate ground truth would provide an ideal testbed for many SLAM techniques.

1.2 Related work

A good introduction to the SLAM problem by Durrant-Whyte and Bailey can be found in [10], which also includes a review of the few known publicly available datasets. Among them, the Victoria’s park [12] has become the most widely employed testbed by the community of 2-d SLAM, having been used in dozens of works, e.g. [11, 15, 20, 18, 21, 23]. Many other datasets can be found in the *Radish repository* [14], most of them consisting of laser scanner and odometry data logs. Nevertheless, no previous dataset contains an accurate ground truth².

Regarding datasets oriented to vision-based SLAM, to the best of our knowledge there is no previous work where a ground truth is associated to the path of the camera or the robot. Another on-going project aimed at SLAM benchmarking is RawSeeds [3]. At the time of writing this article, there are no released datasets yet.

1.3 Contribution

The present work makes two major contributions to the field of mobile robotics.

Firstly, the release of a collection of outdoor datasets with data from a large and heterogeneous set of sensors comprising color cameras, several laser scanners, precise GPS devices and an inertial measurement unit.

This collection provides a unified and extensive testbed for comparing and validating different solutions to common robotics applications, such as SLAM, video tracking or the processing of large 3-d point clouds. The datasets comprise accurate sensor calibration information and both raw and post-processed data (such as the 3-d point clouds for each dataset). The format of all the data files is fully documented and open-source applications are also published to facilitate their visualization, management and processing.

Secondly, we also present a methodology for obtaining a complete 6-d centimeter-accuracy ground truth estimation, among with its associated uncertainty bound. We also discuss innovative auto-calibration methods for some of the sensors, which virtually discard human errors in the manual measurement of the sensor positioning on the vehicle. Additionally, we also introduce a method to measure the consistence of the ground truth, which confirms the accuracy of our datasets.

To the best of our knowledge, there is no previous work with such an accurate ground truth in full 6-d. Visual SLAM techniques are clearly the best candidates to be tested against the presented datasets, although our work may be also applicable to 2-d SLAM techniques due to the existence of two horizontal laser scanners and the planar trajectories followed in some of the datasets.

2 Vehicle description

To carry all the measurement devices, we employed an electric buggy-typed vehicle (see Figure 1) that can navigate through different environments with the appropriate load and autonomy capabilities. The major benefit of using an electric vehicle, besides of the slight environmental impact, is the avoidance of the inherent vibrations of standard cars.

The vehicle was modified and adapted for a more suitable arrangement of the sensors. Specifically, we designed and built a rigid top structure which allowed a robust and easy-to-modify assembly of most of the employed devices.

In total, we placed twelve sensors on the vehicle of such heterogeneous types as laser scanners, inertial measurement units, GPS receivers (both consumer-grade DGPS and GPS-RTK) and color cameras.

Figure 1 (b) shows a descriptive illustration with vehicle sensor positions, and Table 1 provides a listing of sensors and their respective 6-d in the vehicle local reference frame.

These poses were initially measured by hand and, most of them, were subsequently optimized through different calibration methods, as will be explained in

² Some datasets include the GPS positioning of the robot, but not its orientation.

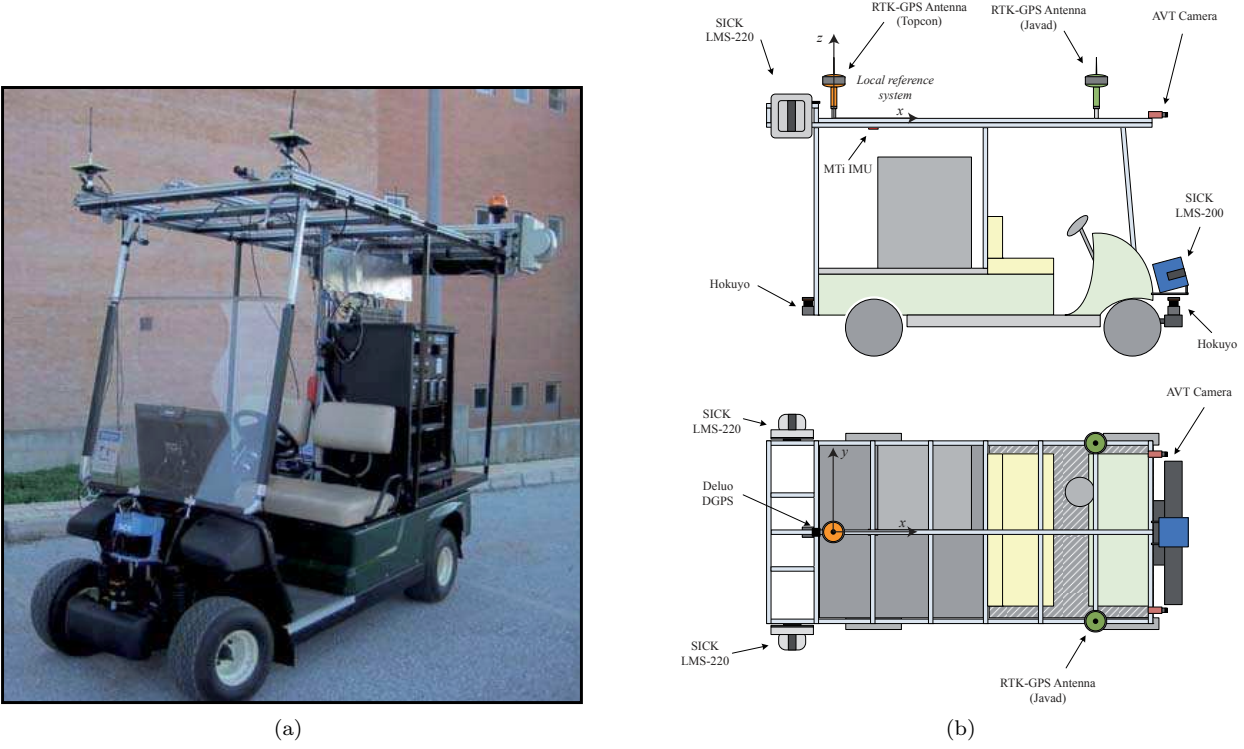


Fig. 1 (a) The buggy-typed electric vehicle employed for grabbing the dataset and (b) a scheme of the sensors positions on the vehicle.

section 5. In the following, we describe each group of sensors.

2.1 Laser Scanners

The vehicle was equipped with three different types of laser scanners: two Hokuyo UTW-30LX and, finally, a SICK LMS-200 (see Figure 2).

The Hokuyo UTW-30LX³ are small, lightweight outdoor laser scanners with a detection range from 0.1 to 30 meters and a 270 deg. field of view. Per manufacturers specifications the sensors provide an angular resolution of 0.25 deg. and achieve an accuracy of 30 mm for

measurements up to 10 meters and 50 mm for higher ranges.

Data transmission between the Hokuyo and the PC is easily accomplished through a USB link, which makes the sensor, together with its lightness and reduced dimensions, quite suitable for mobile robotics applications.

The two Hokuyo laser scanners were located in the front and the rear of the vehicle, respectively, and scanning the environment in a plane parallel to the ground.

On the other hand, SICK⁴ laser scanners are heavy and robust devices widely employed in robotics and industrial applications. Their performance is sensibly higher than Hokuyo's and, furthermore, they are highly configurable.

In our dataset collection, the LMS-200 was configured to measure up to a maximum range of 80 meters with an accuracy of 40 mm (as reported by the manufacturer), whereas the accuracy of the LMS-221 device is 5 mm in a measurement range from 0.8 to 32 meters. Both models provide an angular resolution of 0.5 deg. and a field of view of 180 deg., narrower than that of Hokuyo scanners. Finally, although both the Hokuyo and the LMS-221 devices are designed to operate outdoors, we found in our experiments that the Hokuyo

³ http://www.hokuyo-aut.jp/02sensor/07scanner/utm_30lx.html



Fig. 2 The three laser scanner models employed for the datasets.

⁴ <http://www.sick.com>

Sensor	x (m)	y (m)	z (m)	yaw (deg.)	$pitch$ (deg.)	$roll$ (deg.)
Rear GPS-RTK	0.000	0.000	0.132	×	×	×
Front Left GPS-RTK	1.729	0.5725	0.115	×	×	×
Front Right GPS-RTK	1.733	-0.5725	0.1280	×	×	×
DGPS*	-0.250	0.000	0.100	×	×	×
Left Camera	2.216	0.430	0.022	-88.43	-2.99	-87.23
Right Camera	2.200	-0.427	0.025	-90.31	-3.53	-86.19
Front Hokuyo*	2.420	0.000	-1.740	0.00	0.00	0.00
Rear Hokuyo*	-0.299	0.084	-1.725	178.81	0.00	0.00
Front SICK LMS-200*	2.278	0.000	-1.565	0.00	-6.84	0.00
Left SICK LMS-221	-0.3642	0.7899	0.0441	90.58	6.82	-89.66
Right SICK LMS-221	-0.3225	-0.8045	-0.0201	-90.33	-2.87	89.85
IMU*	×	×	×	0.00	0.00	0.00

*Sensor pose is not optimized. × (irrelevant or not applicable).

Table 1 Summary of the sensor 6-d poses.

laser scanners are more sensitive to sunlight and, therefore, prone to higher errors when operating outdoors.

The LMS-200 sensor was located at the front of the vehicle, scanning a plane slightly tilted upwards, whereas the LMS-221 laser scanners were placed at the sides of the vehicle such as their scans become perpendicular to the ground plane. This layout allows the vehicle to scan objects located at its sides when navigating.

2.2 Cameras

Image grabbing was performed by means of two CCD color cameras (AVT Marlin F-131C model⁵), which can capture up to 1280x1024-size images at a maximum frame speed of 25 fps and transfer them to the PC via a standard firewire link.

In order to properly adjust the devices according to light conditions, some of their features, such as *shutter*, *bright*, or *color mode* can be easily configured by software, while *focus* and *aperture* are manually controlled. The images within the presented datasets were captured at 7.5 fps with a dimension of 1024x768 pixels, being offered sets of both raw and rectified images for each dataset.

The two AVT cameras were mounted on the top structure of the vehicle with their optical axes pointing forward.

To properly rectify the grabbed images, the cameras need to be calibrated, an issue addressed in Section 5.4.

2.3 Inertial measurement unit (IMU)

We employed a MTi device from xSens⁶ (see Figure 3) to provide a source of inertial raw measurements.

The convention followed in this paper for 3-d orientations is to consider them as the sequence of rotations *yaw*, *pitch* and *roll* around the z , y and x axes, respectively, being positive angles performed counter-clockwise.

This device is a miniature, gyro-enhanced, 3-axis IMU, which is both powered and communicated through a USB link. It contains gyroscopes, accelerometers and magnetometers in 3-d which are combined through an Extended Kalman Filter (EKF) to provide 3-d orientation data at a maximum rate of 100 Hz.

According to the manufacturer specifications, the MTi device achieves a static accuracy of 0.5 deg. in attitude (both *pitch* and *roll*) and 1.0 deg. in heading (*yaw*) measurements, whereas in dynamic situations this performance degrades up to 2.0 deg. RMS. The angular resolution of this device is 0.05 deg., while the full scale of the onboard accelerometers and gyroscopes is 5g and 300 deg/s, respectively.

This sensor was also fixed to the top structure of the vehicle in order to move as a rigid solid with the GPS receivers, which define the local coordinates frame. Note that knowledge about the location of this sensor on the

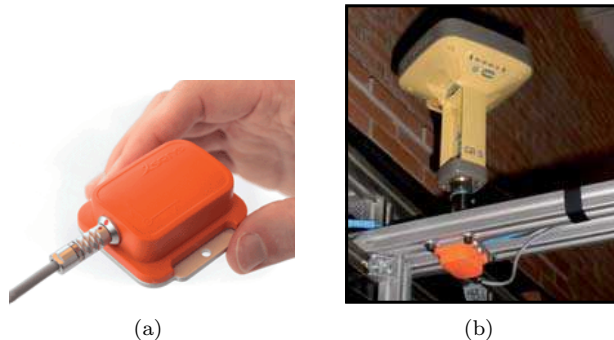


Fig. 3 (a) The MTi IMU model by xSens and (b) its mounting point on the vehicle.

⁵ <http://www.alliedvisiontec.com/avt-products/cameras/marlin.html>

⁶ http://www.xsens.com/en/products/machine_motion/mti.php?

vehicle is not relevant since it only provides orientation measurements.

2.4 GPS devices

Global Positioning System (GPS) has become the most reliable system for land surveying and vehicle positioning due to the high accuracy it can provide and the worldwide spreading of GPS receivers.

In short, the principle of operation of normal GPS systems consists of the trilateration of the distances between a mobile receiver (rover) and a constellation of satellites as illustrated in Figure 4(a). These satellites transmit microwaves signals containing a pseudo-random code (PRC) that is compared by the rover with a stored local copy. The existing delay between the received and the local code determines the actual distances between the receiver and the satellites and, therefore, allows the mobile unit to pinpoint its position. However, normal GPS systems are prone to some error sources such as receiver clock inaccuracies or signal alteration due to atmospheric conditions, and, therefore, provide a relatively low accuracy for mobile robots localization (about 3 meters under ideal conditions).

Differential GPS (DGPS) technique overcomes some of the major limitations and error sources of the normal GPS system by setting a static reference station with a known fixed position which re-broadcast via radio the differential corrections according to the area conditions – see Figure 4(b). These corrections allow the rovers in the area to localize themselves with a higher accuracy (typically tens of centimeters). Normally, DGPS reference stations have large coverage areas (in the order of kilometers) to provide service to as many receivers as possible, which involves a loss of accuracy proportional to the distance to the reference station.

Real-Time-Kinematics (RTK) means an improvement to the DGPS technique which employs both the reference station corrections and the carrier phase of the GPS satellite signals to achieve up to a centimeter level of accuracy. Moreover, GPS-RTK systems usually arrange their own reference stations which can be placed much closer to the rovers than those employed for standard DGPS systems, as shown in Figure 4(c). In this state-of-the-art technique, the GPS receiver gages the distance to the satellites by adjusting the received and the local copy of the signal not only by comparing the PRC but also the phase of the signals, thus leading to a much more fine delay estimation and, therefore, to an improved localization (up to 1 cm of accuracy).

GPS-RTK devices can operate in two different modes: RTK-float and RTK-fixed. In the former, the RTK reference station has not enough information from the

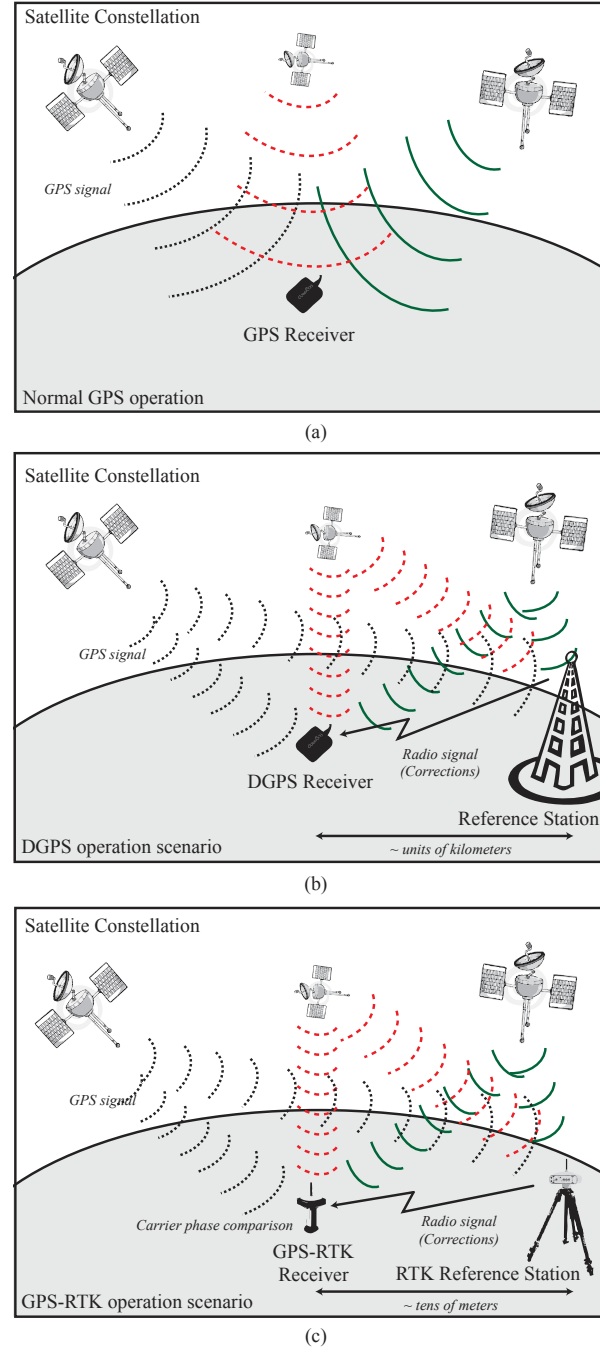


Fig. 4 GPS scenarios. (a) Normal GPS operation, (b) DGPS operation and (c) Real-Time-Kinematics GPS operation.

satellite constellation to precisely determinate its static position, leading to significant errors in rovers localization, while the latter is only attainable when the number of visible satellites and the quality of their signals allow a total disambiguation in its positioning.

We placed four different GPS receivers in our vehicle: one consumer-grade DGPS device and three RTK-GPS (two Javad Maxor and one Topcon GR-3) which



Fig. 5 The GPS devices on the vehicle. (a) Low cost Deluo DGPS, (b) Javad Maxor GPS-RTK and (c) Topcon GR-3 GPS-RTK.

constitute a solid frame for establishing a reliable 6-d ground truth estimation (see Figure 5).

The two Maxor-GGDT devices from Javad⁷ are dual frequency GPS receivers which provide RTK-level-of-accuracy measurements at a rate of 4 Hz through a RS-232 link. These receivers are associated to a pair of MarAnt+ antennas which were mounted, as separated as possible, at the front of the vehicle top structure, as can be seen in Figure 1 (b).

On the other hand, the Topcon GR-3⁸ device is a robust, state-of-the-art positioning receiver which can combine signals from the three existing satellite constellations: American GPS, Russian GLONASS and European Galileo, thus ensuring an optimal coverage 24h a day achieving a high performance. Positioning data is measured and transmitted to the PC at a rate of 1 Hz. This device was placed at the rear of the top structure.

It is important to remark that Javad and Topcon devices are connected to different reference stations as they are configured to receive RTK corrections from their own stations through different radio channels. This caused the positioning readings to be biased with constant offsets depending on the specific measuring device. This offset can be estimated through a least squares-error minimization algorithm, which will be explained in Section 4.2.

Finally, the Deluo⁹ DGPS is a low-cost, consumer-grade, compact and portable receiver extensively employed for extending PDAs and laptops with self-localization capabilities. Its performance and accuracy are sensibly poorer than in the RTK-GPS sensors but its measurements may be suitable for testing and developing rough localization applications which are intended to be accessible to a high amount of users. For example, it could be taken into account together with the cameras to test large-scale SLAM approaches. Section 5.2 presents a brief evaluation of the Deluo DGPS performance.

3 Data collection

This section describes the emplacements of the six presented datasets. Three of them were collected at the parking of the Computer Science School building while the other three were located at the Campus boulevard of the University of Málaga, as depicted in Figure 6.

From now on, and for clarity purposes, the datasets will be denoted by the combination of a prefix, identifying its corresponding emplacement (PARKING or CAMPUS) and two-letters mnemonic indicating a distinctive feature that makes it easily identifiable. In five of the six datasets, this mnemonic denotes the number of complete loops performed by the vehicle (e.g. 0L for zero loops) whereas in the remaining one (CAMPUS-RT), it indicates that the trajectory is a round-trip.

Next, we describe more extensively the most relevant characteristics of the two groups of datasets.

3.1 Parking datasets

The three *parking* datasets are characterized for almost planar paths through a scenario which mostly contains trees and cars. These datasets present more loops than the *campus* ones, which makes them specially interesting for testing SLAM approaches.

The lack of planar surfaces, such as buildings, and the presence of dynamic objects such as moving cars and crossing people, make this emplacement particularly interesting for computer vision applications (e.g. segmentation or object visual tracking). Finally, the presence in this group of datasets of nearby objects in the images, facilitates the estimation of the camera displacement in common vision-based solutions while the detection of far objects leads to a better estimation of the rotation [5,6].

Figures 7–9 contain visual descriptions of the *parking* datasets, showing the 3-d point clouds (subfigures (a), (c) and (e)) generated through the projection of the vertical laser scans from their estimated pose at each time step. The methodology for the estimation of the sensors trajectories will be described later on in Section 6.3.

On the other hand, subfigures (b) illustrate aerial views of the whole vehicle trajectory where it has been highlighted the start and end points of the paths.

Finally, we depict in subfigures (d) a summary of the operation modes for the three GPS-RTK receivers during the whole data collection process (please, refer to section 2.4 for a descriptive review of GPS operation modes). Notice that we can estimate a reliable ground truth only when the three devices are simultaneously

⁷ <http://www.javad.com/jns/index.html>

⁸ <http://www.topconeurope.com>

⁹ <http://www.deluogps.com>

operating in RTK-fixed mode, thereby achieving the highest accuracy of the 6-d pose vehicle estimation.

Please, note as well that we have considered the starting point of the dataset PARKING-0L as the origin of the Cartesian reference system for all the presented datasets, as can be seen in Figure 6.

3.2 Campus datasets

In general, the *campus* datasets include fairly large loops, specially suitable for validating large-scale SLAM algorithms. Besides, they also contain long straight trajectories which may be a proper testbed for visual odometry methods. Approaches for other vision-based applications such as automatic detection of overtaking cars or dynamic objects tracking are also good candidates for being evaluated by employing these datasets.

Their visual descriptions are shown in Figures 10-12 with identical structure as explained above.

3.3 Summary of dataset applications

Table 2 shows a summary of some datasets properties and an evaluation of their suitability for being employed in a set of common robotics and computer vision applications. This should be understood as a recommendation, related to the own special characteristics of each dataset as, for example, the number of relevant sensors, the planar nature of the movement, or the presence of loop closures.

3.4 Software

The vehicle is equipped with sensors of quite different types, each generating data at different rates. Thus, the software intended to grab the data logs must be capable of dealing with asynchronous sensor data. For this purpose, our data log recording application, named *rawlog-grabber*, creates one thread for each individual sensor on the robot. Each thread collects the data flow from its corresponding sensor, and converts it into discrete entities, or *observations*. Different kinds of sensors produce different observation objects.

At a predetermined rate (1Hz in our case), the main thread of the logger application collects the incoming observations from all sensors, which are then sorted by their timestamps and dumped into a compressed binary file. Documentation about the format of these files, as well as the source-code of the grabber and some data viewer applications are published as part of the Mobile Robot Programming Toolkit (MRPT) [2].

4 Derivation of the path ground truth

In this section we address one of the major contributions of this work: a detailed description of how to compute a ground truth for the 6-d path of the vehicle from GPS readings. Section 6 will derive a bound for the uncertainty of this reconstructed path.

4.1 Coordinates transformation

We will focus first on the problem of tracking the position of a *single* GPS receiver, that is, we are interested in the 3-d coordinates (x, y, z) of a particular point. In our situation, these points are the centers of the GPS antennas – see Figure 1.

GPS receivers provide datums through three parameters: longitude, latitude and elevation. To fully exploit the precision of RTK receivers, these coordinates must be interpreted exactly using the same system the GPS network uses, the World Geodetic System (WGS)-84 reference ellipsoid. This coordinate framework has been optimized such as its center matches the Earth center of mass as accurately as possible. We briefly explain next the meaning of the three coordinates in WGS-84.

The *longitude* datum states the angular distance from an arbitrarily defined meridian, the International Reference Meridian, just a few arc-seconds off the prior Greenwich’s observatory reference. The *geodetic latitude* is the other angular coordinate, which measures the angle between the equatorial plane and the normal of the ellipsoid at the measured point. Note that this is not exactly equal to the angle for the line passing through the Earth center and the point of interest (the *geocentric* latitude [4]). Finally, the *elevation* represents the height over the reference geoid.

We now review the equations required to reconstruct local Cartesian coordinates, the natural reference system employed in SLAM research, from a sequence of GPS-collected data. Let D_i be the datums of the i ’th GPS reading, comprised of longitude α_i , latitude β_i and elevation h_i :

$$D_i = \begin{bmatrix} \alpha_i \\ \beta_i \\ h_i \end{bmatrix} \quad (1)$$

The geocentric Cartesian coordinates of this point, G_i , can be obtained by [16]:

$$G_i = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \begin{bmatrix} (N_i + h_i) \cos \beta_i \cos \alpha_i \\ (N_i + h_i) \cos \beta_i \sin \alpha_i \\ (N_i \cos^2 \beta_i + h_i) \sin \beta_i \end{bmatrix} \quad (2)$$

	Length	LC	vSLAM	2D SLAM	vOdometry	Dyn. Obj. Seg.
PARKING-0L	524 m.	×	✓	✓	✓	×
PARKING-2L	543 m.	✓	✓	✓	✓	×
PARKING-6L	1222 m.	✓	✓	✓	✓	✓
CAMPUS-0L	1143 m.	×	✓	×	✓	✓
CAMPUS-2L	2150 m.	✓	✓	×	✓	✓
CAMPUS-RT	776 m.	×	✓	×	✓	✓

Table 2 Datasets lengths and recommended applications. **Legend:** LC (*Loop Closure*), vSLAM (*Visual SLAM*), vOdometry (*Visual Odometry*), Dyn. Obj. Seg. (*Dynamic Objects Segmentation*)



Fig. 6 Vehicle trajectories for the six grabbed datasets, where P and C stand for PARKING and CAMPUS, respectively. The global reference system represents the origin in our transformed Cartesian coordinate system for all the datasets.

with the radius of curvature N_i computed from the semimajor axis a and the angular eccentricity ϵ as:

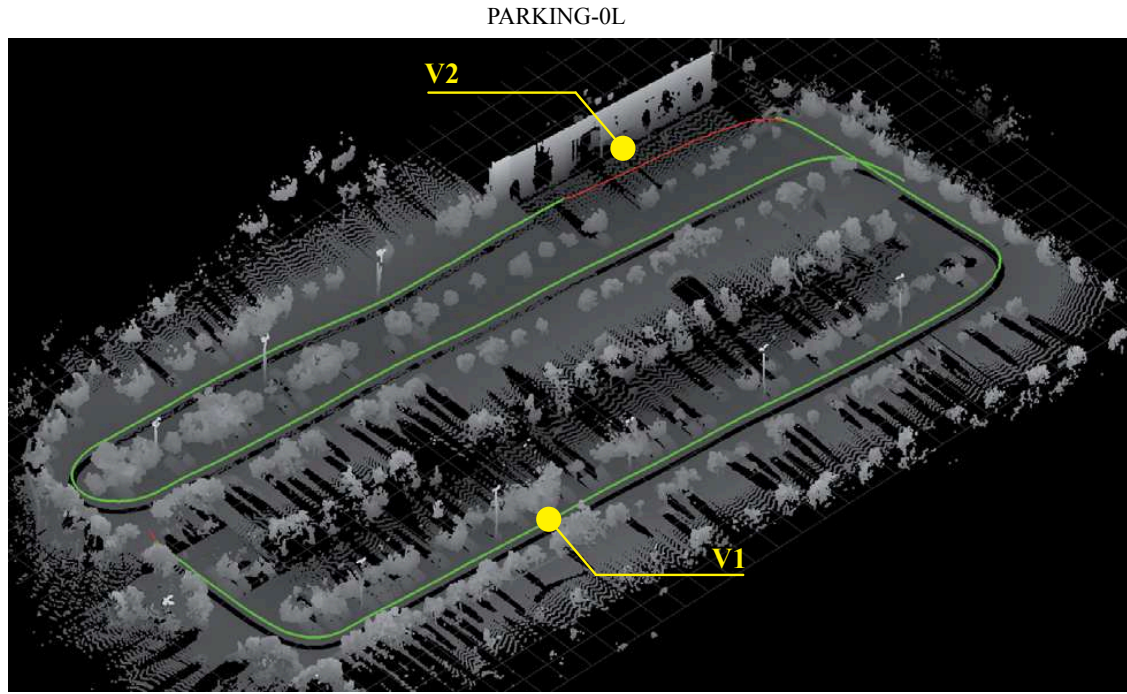
$$N_i = \frac{a}{\sqrt{1 - \sin^2 \epsilon \sin^2 \beta_i}} \quad (3)$$

At this point we have assigned each GPS readings a 3-d location in rectangular coordinates. However, these coordinates are of few practical utility due to two problems: the large scale of all the distances (the origin is the center of the Earth), and the orientation of the XYZ axes – refer to Figure 13(a). We would rather desire a coordinate transformation where coordinates are local to a known point in the environment and the axes have a more convenient orientation, with the XY plane being

horizontal and Z pointing upward, as illustrated in Figure 13(b). This kind of reference system is called ENU (East-North-Up) and is commonly used in tracking applications, as opposed to the so far described Earth-Centered Earth-Fixed (ECEF) system [16].

Unlike other previous methods such as [9], our change of coordinates is not an approximation but an exact simple rigid transformation, i.e. computed accurately without approximations.

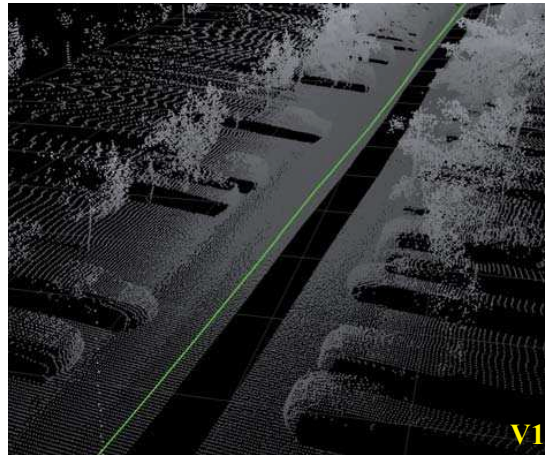
This change of coordinates can be described as the mapping of a 3-d point G_i into local ENU coordinates L_i relative to another point R , with rotation represented by the three new orthogonal base vectors u (East),



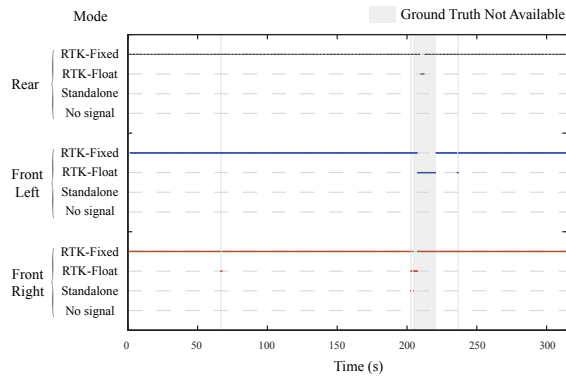
(a)



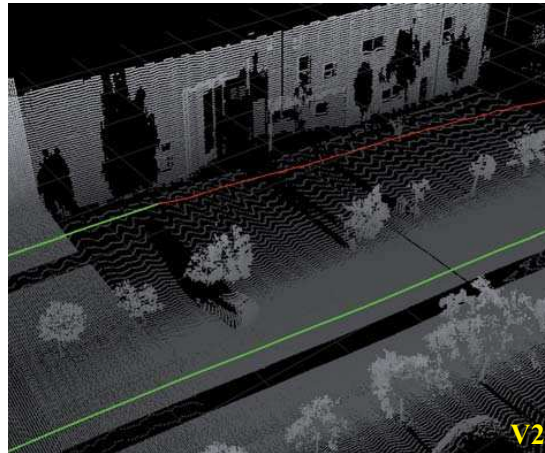
(b)



(c)



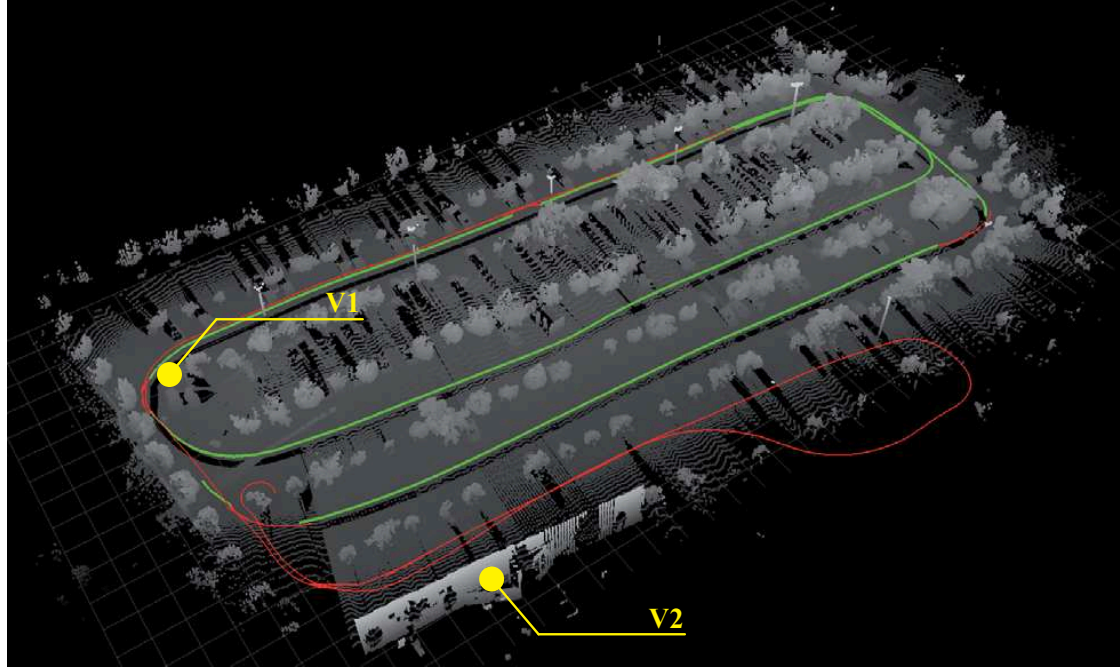
(d)



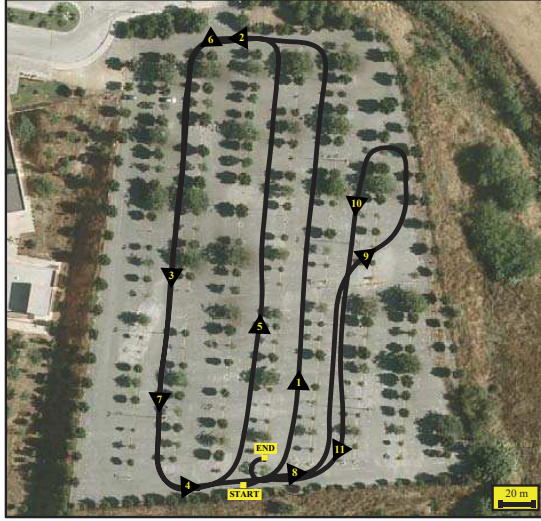
(e)

Fig. 7 PARKING-0L dataset. (a) 3-d projected points for the lateral laser scanners and vehicle path. (c,e) Zoom of zones V1 and V2. (b) Top-view of the vehicle path and (d) status (modes) of the 3 GPS devices during the experiment.

PARKING-2L



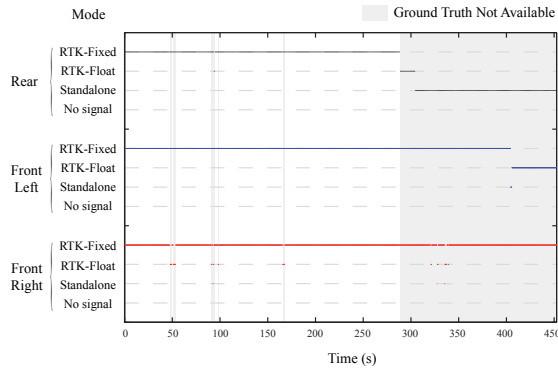
(a)



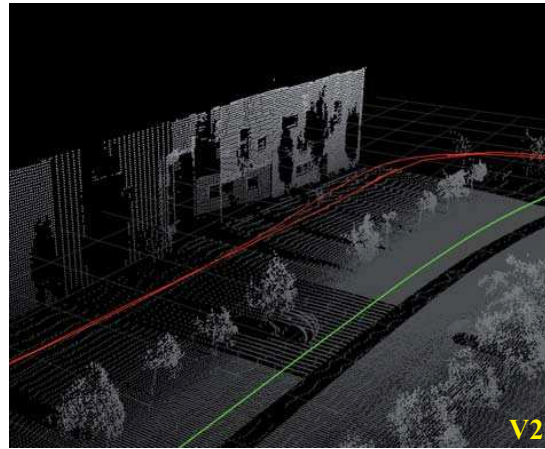
(b)



(c)



(d)



(e)

Fig. 8 PARKING-2L dataset. (a) 3-d projected points for the lateral laser scanners and vehicle path. (c,e) Zoom of zones V1 and V2. (b) Top-view of the vehicle path and (d) status (modes) of the 3 GPS devices during the experiment.

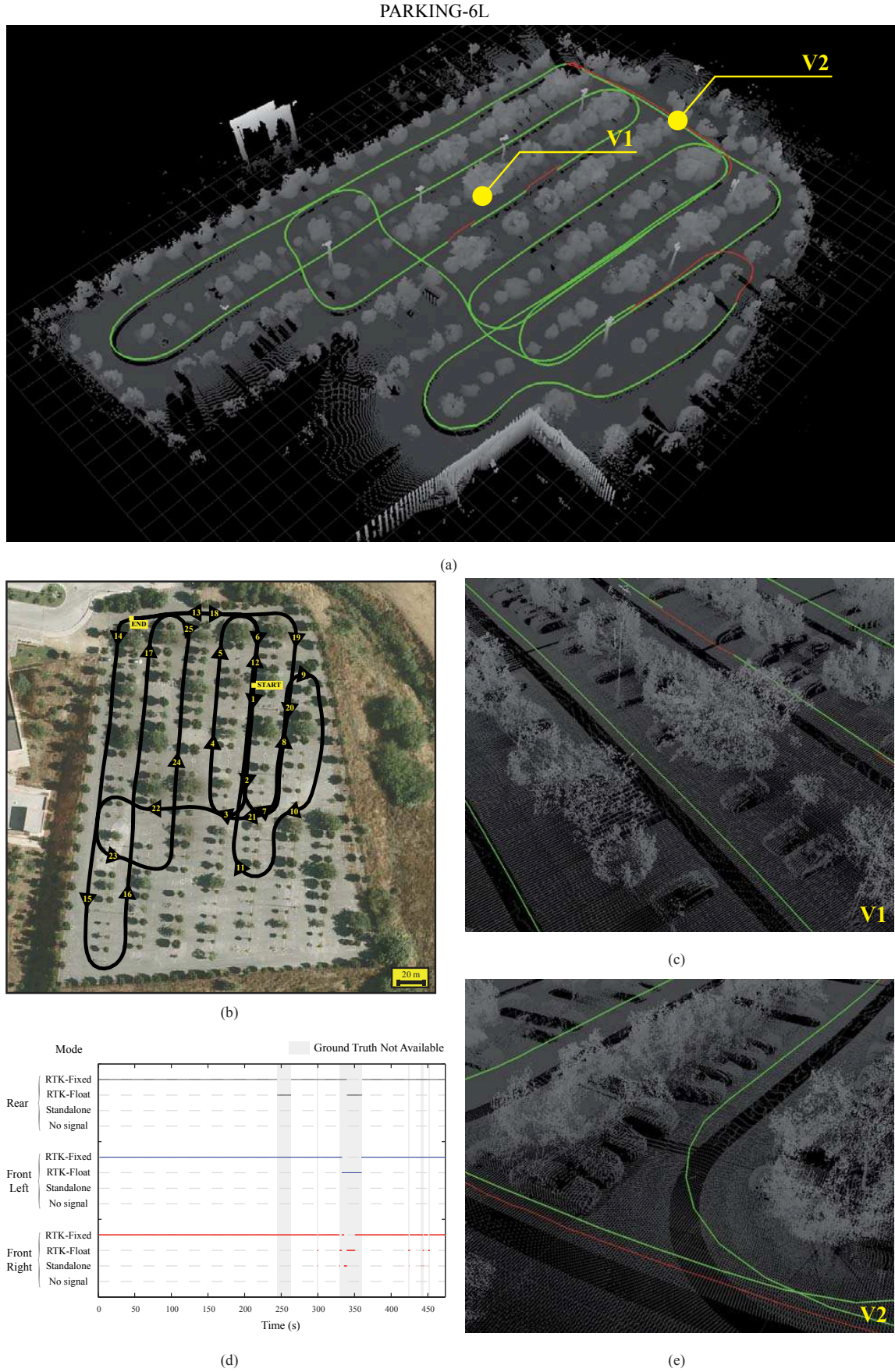
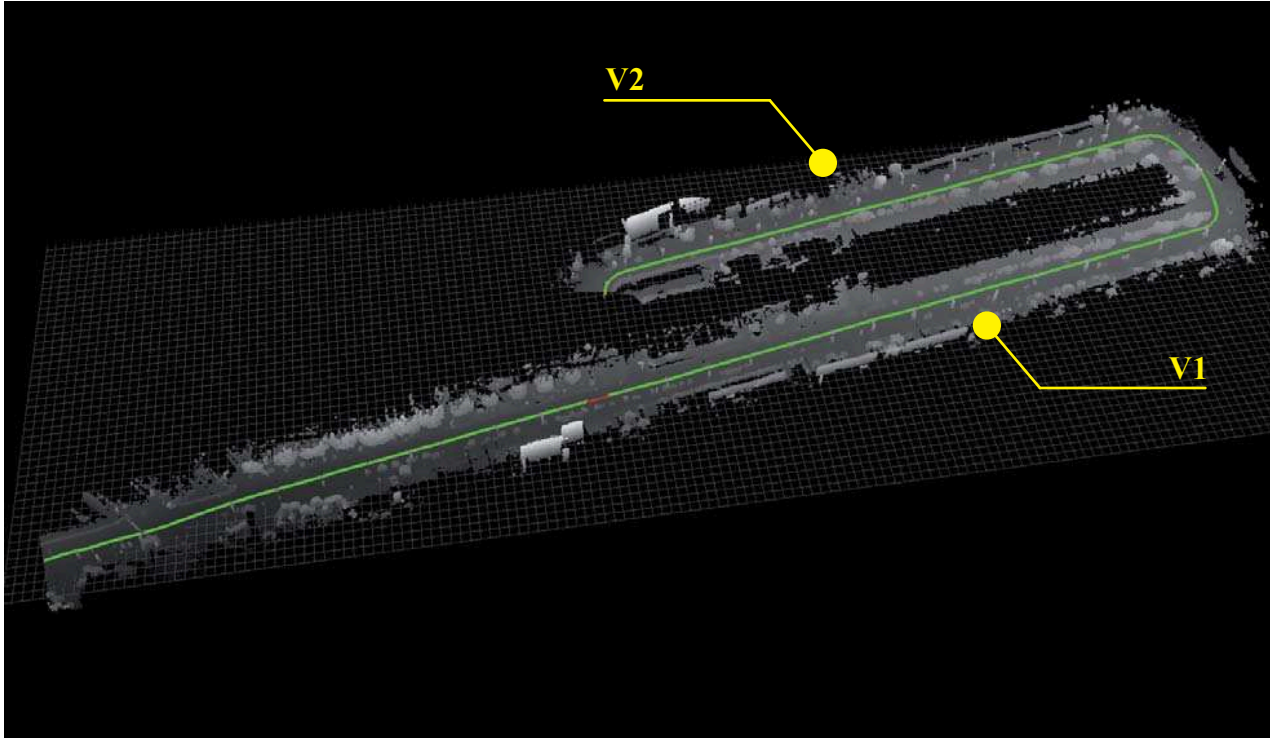


Fig. 9 PARKING-6L dataset. (a) 3-d projected points for the lateral laser scanners and vehicle path. (c,e) Zoom of zones V1 and V2. (b) Top-view of the vehicle path and (d) status (modes) of the 3 GPS devices during the experiment.

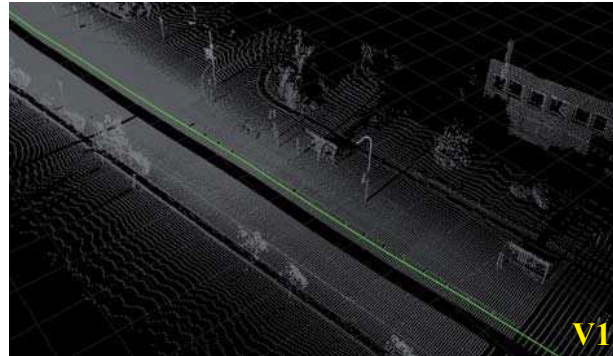
CAMPUS-0L



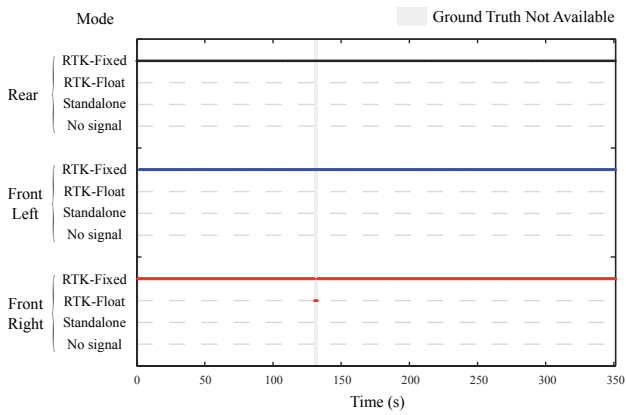
(a)



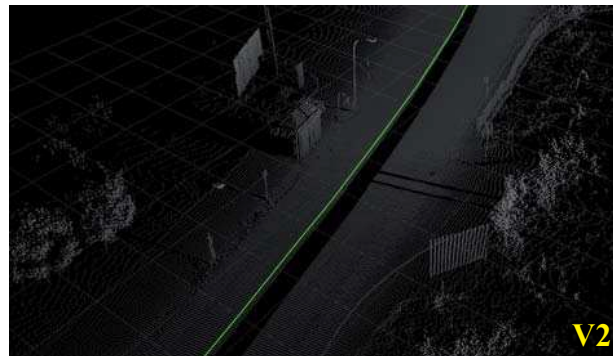
(b)



(c)



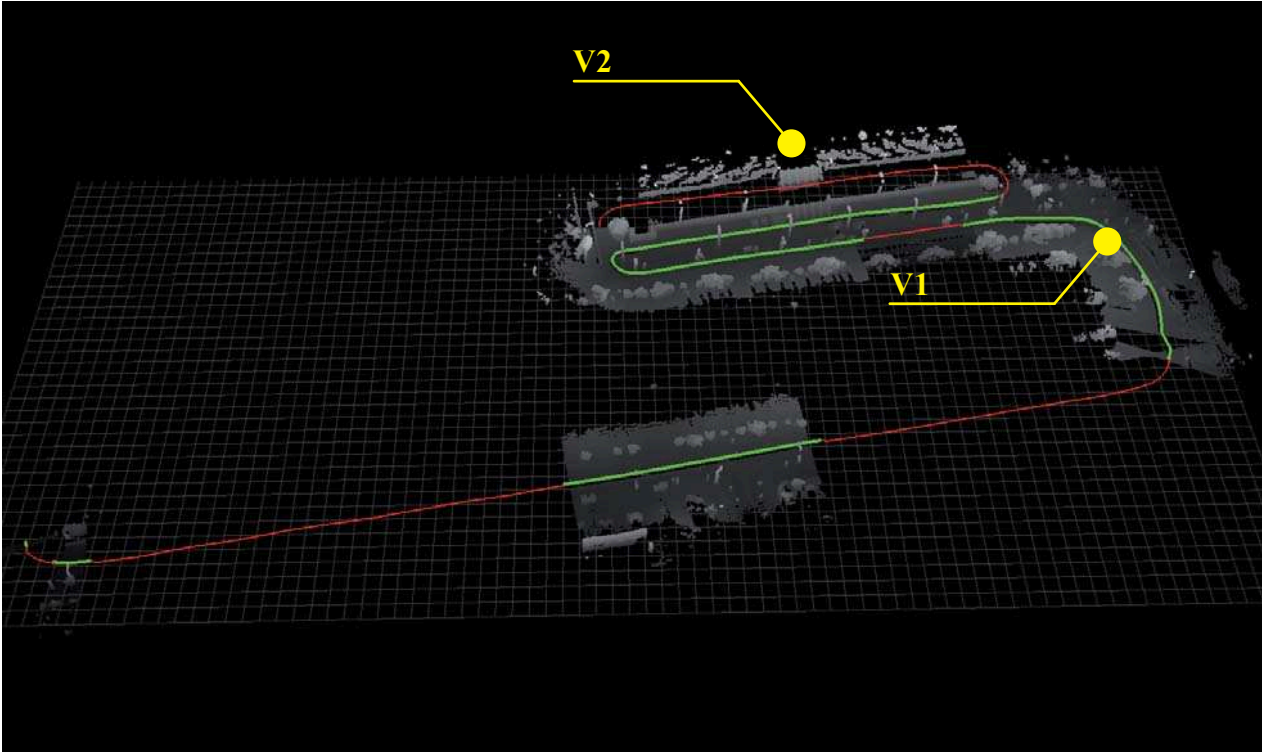
(d)



(e)

Fig. 10 CAMPUS-0L dataset. (a) 3-d projected points for the lateral laser scanners and vehicle path. (c,e) Zoom of zones V1 and V2. (b) Top-view of the vehicle path and (d) status (modes) of the 3 GPS devices during the experiment.

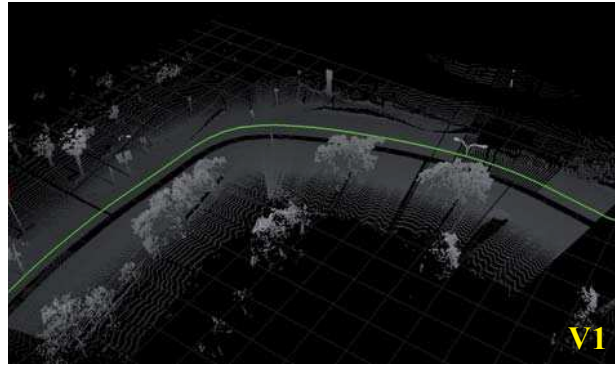
CAMPUS-RT



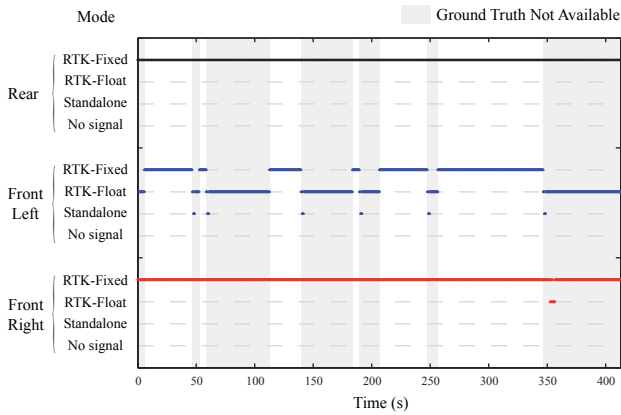
(a)



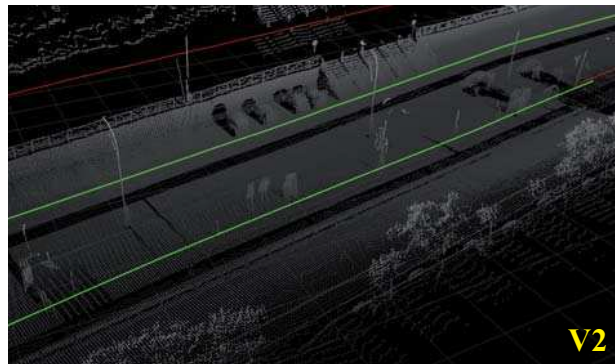
(b)



(c)



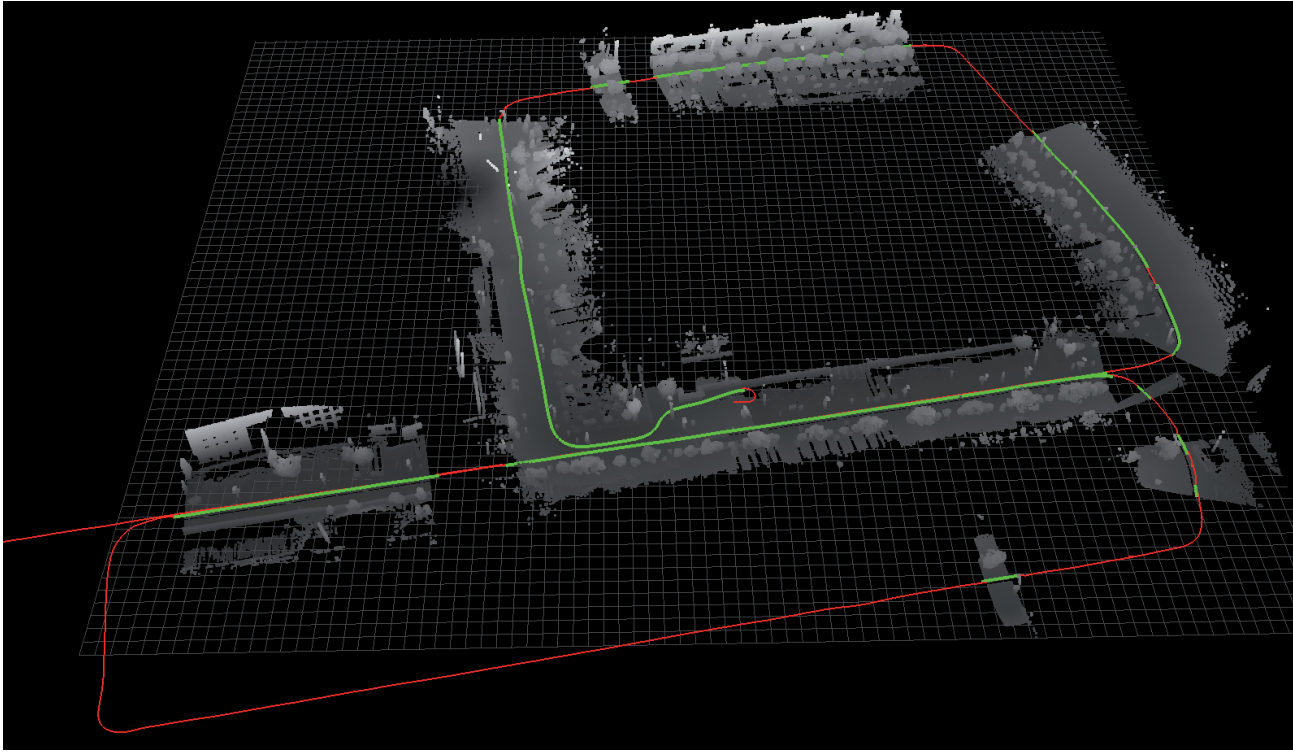
(d)



(e)

Fig. 11 CAMPUS-RT dataset. (a) 3-d projected points for the lateral laser scanners and vehicle path. (c,e) Zoom of zones V1 and V2. (b) Top-view of the vehicle path and (d) status (modes) of the 3 GPS devices during the experiment.

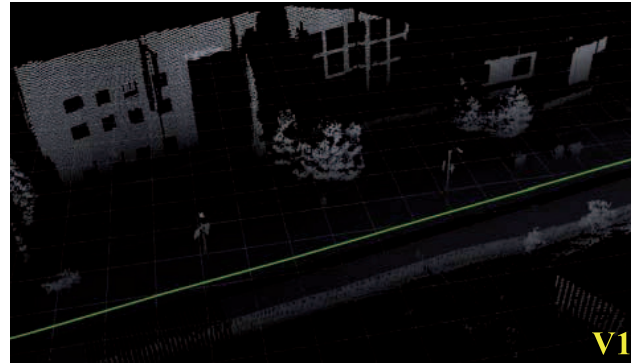
CAMPUS-2L



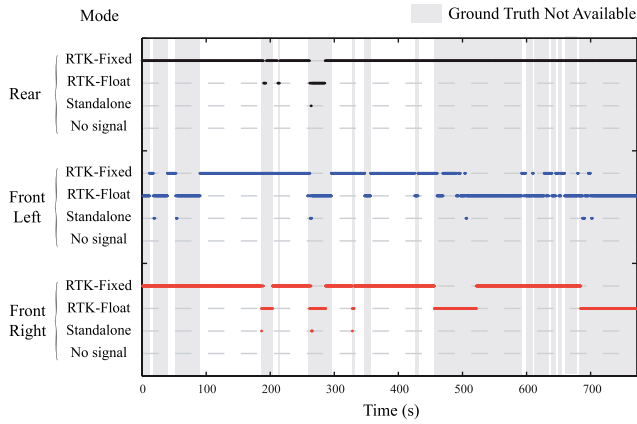
(a)



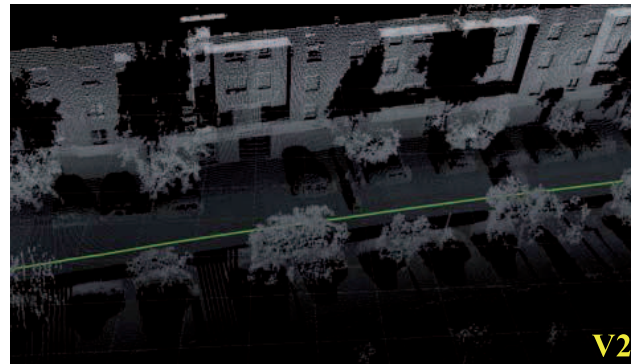
(b)



(c)



(d)



(e)

Fig. 12 CAMPUS-2L dataset. (a) 3-d projected points for the lateral laser scanners and vehicle path. (c,e) Zoom of zones V1 and V2. (b) Top-view of the vehicle path and (d) status (modes) of the 3 GPS devices during the experiment.

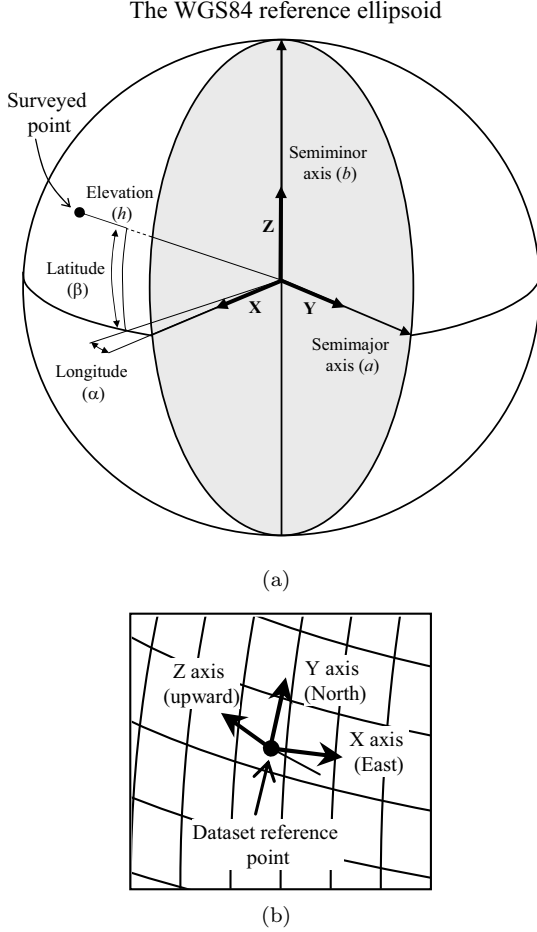


Fig. 13 (a) A schematic representation of the variables involved in the WGS84 reference ellipsoid used for GPS localization. We define a more convenient local Cartesian coordinate system with the origin at an arbitrary location, as shown in (b).

v (North) and w (up)¹⁰. Mathematically, the operation can be written down using homogeneous matrices as:

$$L_i = \begin{bmatrix} u & v & w & R \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} G_i \quad (4)$$

$$= \begin{bmatrix} u^\top & -u^\top R \\ v^\top & -v^\top R \\ w^\top & -w^\top R \\ 0 & 1 \end{bmatrix} G_i$$

However, we have found that a direct implementation of the equation above suffers of unacceptable in-

¹⁰ In the published datasets, coordinates use an *up* vector that follows the same direction than the line passing through the Earth center and the reference point. In the literature, this vector is often defined as perpendicular to the ellipsoid (very close but not exactly equal to our *up* vector). Thus, strictly speaking, ours are not ENU coordinates but a slightly rotated version. Nevertheless, this does not affect at all the accuracy of the obtained coordinates.

accuracies due to numerical rounding errors with standard 64-bit floating point numbers. These errors arise in the multiplications of numbers in a wide dynamic range, i.e. the orthogonal vectors (in the range $[0, 1]$) and the geocentric coordinates (with a order of 10^6).

As a workaround, we propose the following rearrangement of the transformation:

$$L_i = \begin{bmatrix} u^\top & 0 \\ v^\top & 0 \\ w^\top & 0 \\ 0 & 1 \end{bmatrix} (G_i - R) \quad (5)$$

which can be easily derived from Eq. (4) but avoids the numerical inaccuracies.

Finally, the geodetic coordinates of the reference point used in all the datasets presented in this work can be found in the next table (see also its representation in the map of Figure 6):

Longitude:	-4.47895882833333 deg.
Latitude:	36.714459075 deg.
Elevation:	38.8887 m.

4.2 Compensation of RTK offsets

At this point we can compute the trajectories of the three RTK GPS receivers. However, as mentioned in Section 2.4, the fact that RTK corrections are taken from different base stations introduce constant offsets in the locations of each device. A part of these offsets is also due to inaccuracies in the initial positioning of the RTK base stations during the system setup.

Let \hat{P}_t^i denote the local Cartesian coordinates of the i 'th GPS receiver, computed as described in the previous section. Our goal is to obtain the corrected coordinates P_t^i :

$$P_t^i(\Delta^i) = \begin{bmatrix} x_t^i \\ y_t^i \\ z_t^i \end{bmatrix} = \hat{P}_t^i + \Delta^i = \begin{bmatrix} \hat{x}_t^i \\ \hat{y}_t^i \\ \hat{z}_t^i \end{bmatrix} + \begin{bmatrix} \Delta_x^i \\ \Delta_y^i \\ \Delta_z^i \end{bmatrix} \quad (6)$$

by means of the offset vectors which are different for each receiver i but constant with time. Let Δ denote the concatenation of all these vectors, such as

$$\Delta = \begin{bmatrix} \Delta^1 \\ \Delta^2 \\ \dots \\ \Delta^i \end{bmatrix} \quad (7)$$

We show next how these parameters can be determined automatically *without any further a-priori known*

	Campus Datasets			Parking Datasets		
	Optimal	RMSE of d_{ij}	RMSE of d_{ij}^2	Optimal	RMSE of d_{ij}	RMSE of d_{ij}^2
$d_{1,2}$	1.8226m	1.16cm	4.23510^{-2}	1.8208m	1.02cm	3.71110^{-2}
$d_{1,3}$	1.8255m	0.89cm	3.32610^{-2}	1.8247m	0.71cm	2.61110^{-2}
$d_{2,3}$	1.1444m	0.81cm	1.84810^{-2}	1.1457m	0.96cm	2.18910^{-2}
Optimal RTK offsets $(\Delta_x^i, \Delta_y^i, \Delta_z^i)$ (meters)						
Δ^1	0, 0, 0			0, 0, 0		
Δ^2	-145.2875, 28.6745, 2.3344			-260.168, 23.8158, -1.64305		
Δ^3	-145.2914, 28.6712, 1.2915			-260.169, 23.8075, -2.74191		

Table 3 Results of the least squares optimizations of GPS parameters.

data, measurements or approximations. This is a crucial point supporting the quality of our subsequently derived ground truth, since our method is insensitive to errors in any manually acquired (i.e. noisy) measurement.

The basis of this automatic calibration procedure is that the three GPS receivers move as a single rigid body, that is, they are robustly attached to the vehicle (refer to Figure 1). From this follows that the distances between GPS receivers (or *inter-GPS distances*) must be constant with time, which allows us to set up the determination of Δ as the following least square optimization problem:

$$\Delta^* = \arg \min_{\Delta} E(\Delta, \mathbf{D}) \quad (8)$$

$$E(\Delta, \mathbf{D}) = \sum_{(i,j)} \sum_t \left(\left| P_t^i(\Delta^i) - P_t^j(\Delta^j) \right| - d_{ij} \right)^2$$

where (i, j) represents all the possible unique pairs of GPS devices ¹¹, and $\mathbf{D} = \{d_{ij}\}$ are the real distances between those pairs. Notice that those distances d_{ij} can be determined by measuring them manually, what for our vehicle gives us:

$$\begin{aligned} d_{rear, front-left} &= 1.79 \text{ meters} \\ d_{rear, front-right} &= 1.79 \text{ meters} \\ d_{front-left, front-right} &= 1.15 \text{ meters} \end{aligned}$$

Obviously, small errors must certainly be present in these values due to the limited accuracy of any practical method for manual measuring. In order to make our calibration independent of those errors, the set of optimal inter-GPS distances \mathbf{D}^* is determined as the result of another optimization problem, taking the values above as the initial guess for the optimization:

$$\mathbf{D}^* = \arg \min_{\mathbf{D}} \left(\min_{\Delta} E(\Delta, \mathbf{D}) \right) \quad (9)$$

¹¹ In our specific case of three receivers, the possible values are (1, 2), (1, 3) and (2, 3), where the indices 1, 2, and 3 correspond to the rear, front-left and front-right receivers, respectively.

We must remark that this is a nested optimization problem, where the inner optimization of Δ was stated by Eq. (8).

Put in words, our approach to the automatic determination of the offsets Δ and inter-GPS distances \mathbf{D} consists of an iterative optimization of the inter-GPS distances, where the error function being optimized is the residual error of another optimization of the RTK offsets, maintaining the distances fixed.

The optimization has been implemented as two nested instances of the Levenberg-Marquardt algorithm [17, 19]. The method exhibits an excellent robustness, in the sense that it converges to the optimal values starting from virtually any set of initial values for Δ . Regarding the execution time, our custom C++ implementation takes about 2 minutes to optimize the data from a representative sample of 600 timesteps.

Since the six datasets, described in Section 3, were collected during two days with the RTK base stations installed at different locations each day, the RTK offset parameter Δ has different values in the *campus* datasets than in the *parking* datasets. On the other hand, the inter-GPS distances \mathbf{D} should remain constant throughout the different datasets.

The optimization results are summarized in Table 3: the average difference between the values of \mathbf{D} in the two independent optimizations is 1.3mm, i.e. in practice both estimates converge to the same point. This is a clear indication of the accuracy of all the 3-d locations given by the sensors, an issue which is quantified later on in this work.

We must highlight again that our method has precisely estimated these distances and the RTK offsets (in the order of hundreds of meters) by simply relying on the rigid body assumption of the GPS receivers.

4.3 Computation of the vehicle trajectory

As a result of the previous section, we now have an accurate 3-d reconstruction of the path followed by each GPS receiver on the vehicle, namely the sequences P_t^i

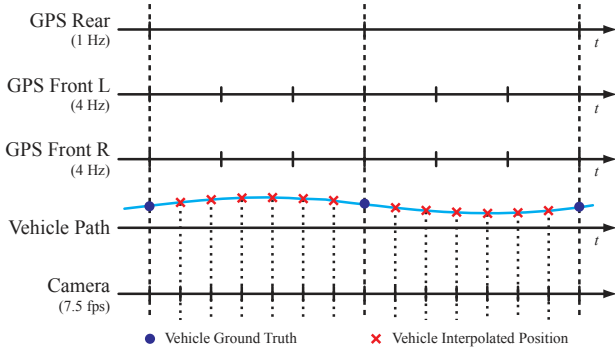


Fig. 14 Vertical ticks represent the timestamps at which the different sensors give their readings. In the case of the GPS receivers, the rear device works at 1Hz while both the front left and right devices have a frequency of 4Hz. Thus the ground truth for the vehicle pose is obtained at 1Hz, the rate at which all three sensors coincide. Since other sensors (in this example, a camera) operate at different rates we need to interpolate the 6-d ground truth to assign a valid pose to each sensor reading.

for the three GPS sensor $i = 1, 2, 3$ and for the sequence of timesteps t .

At this point it must be pointed out that the timing reference used by each GPS receiver is very accurately synchronized to the others since they are all synchronized with the satellite signal. Therefore, when the timestamp of datum readings from different receivers coincide we can assume that all the GPS locations were measured exactly at the same time, with timing errors negligible for the purposes of this work.

The synchronization of the data is a prerequisite for our goal of computing the 6-d ground truth of the vehicle path, since three simultaneous 3-d measurements unequivocally determine the complete 6-d pose of a rigid body, in our case the vehicle. Therefore, ground truth is available when the data from all the three receivers coincide in time. As illustrated in Figure 14, this happens with a rate of 1Hz in our system.

Regarding the computation of the sequence of 6-d vehicle poses v_t from the GPS locations P_t^i , we can set up the following optimization problem which gives the optimal solution at each timestamp t :

$$\hat{v}_t = \arg \min_v \sum_i (P_t^i - [v \oplus p_i])^2 \quad (10)$$

with p_i stating the location of each receiver on the vehicle local coordinate framework (refer to Figure 1), and \oplus being the pose composition operator [22]. This can be regarded as a problem of matching three pairs of corresponding 3-d points. A widely employed solution to this generic problem is the Iterative Closest Point (ICP) algorithm [1]. In turn, we propose to apply the closed-form solution derived by Horn in [13] since in our case there are no uncertain data associations (the

main issue solved by ICP). Other interesting characteristics of [13] are the closed-form nature of the solution and that providing an initial guess of the solution is unnecessary.

One advantage of stating the problem of recovering the vehicle path as in Eq. (10) is that GPS sensors can be freely positioned at arbitrary locations on the vehicle, e.g. there is no assumption about them being disposed in orthogonal directions. The process to automatically refine the local coordinates p_i is discussed later on in Section 5.1. We must remark that a minimum of three GPS devices are required to obtain instantaneous ground truth measurements, but our method can trivially incorporate any larger number of devices with the purpose of reducing the overall reconstruction error.

4.4 Interpolation of the 6-d vehicle trajectory

The process described in Section 4.3 gives us the desired ground truth for the vehicle trajectory, one of the main goals of this work.

However, there are two tasks that force us to interpolate the 1Hz ground truth. Firstly, vision SLAM methods usually reconstruct the 6-d path of the *cameras*, not the *vehicle*. Therefore, in order to measure the quality of a given SLAM method under evaluation, a ground truth for the path of the cameras (or in general any other sensor, e.g. laser scanners) must be available. This is why the vehicle path must be interpolated, since in general the timestamps of other sensors will not coincide with those of the GPS readings, as shown in Figure 14. Secondly, our goal of building accurate point clouds from the laser scanner data also requires the precise locations of the scanners with time, again requiring the vehicle pose at instants of time that require interpolation.

A priori, a potentially promising technique for carrying on this interpolation would be spline fit [8], which results in smooth interpolated curves with continuous derivatives that exactly pass through all the input points to interpolate.

The consequences of this last property of splines in our specific case require a closer examination. Each of the six dimensions of the vehicle pose, 3-d for Cartesian coordinates and 3-d for the angles, follows characteristic and distinctive variations with time. For example, the coordinates of the planar locations (x and y) are specially suited for being approximated by splines since the vehicle movement over the road tends to describe smooth arcs. On the other hand, the pitch and roll angles are typically around zero and disregarding the small part caused by real rocking of the vehicle,

variations are caused by the noise (errors) in the 3-d coordinates of the RTK receivers. In those cases, splines typically magnify this error making it an unacceptable choice.

In our datasets, we finally chose to interpolate the vehicle x and y coordinates using splines, and the rest of dimensions using a least square linear fit of the four closer ground truth points. Once the pose of the vehicle can be estimated at any arbitrary instant t , the sensor pose is obtained by 6-d pose composition with the local sensor location (see Table 1), for example, using homogeneous coordinates. This method has led to accurate reconstructed paths, as can be visually verified with the maps discussed in Section 7.1. In Section 6.3 we go back to the issue of sensor paths and their uncertainty.

5 Calibration of sensors

The 6-d positioning of each sensor in vehicle local coordinates is as important as the ground truth of the vehicle trajectory itself. By means of carefully manual measuring we can obtain a first estimate of these parameters associated to every sensor onboard. The purpose of this section is to discuss automatic calibration methods for some of the sensors used in the datasets.

It must be noted the hierarchical nature of our calibration methodology: first of all, the inter-GPS distances were automatically calibrated through the least square problem stated in Eq. 9. As explained in Section 4.2, that solution is free of human measuring errors. In the subsequent sections these optimal inter-GPS distances \mathbf{D}^* will be used to first refine the Cartesian coordinates of the GPS receivers. Next, those coordinates are the basis for an accurate determination of the vehicle path ground truth, which in turn is indirectly used later to optimize the positions of the vertical laser scanners and the cameras. The final result of all this process can be seen in Table 1.

5.1 Positioning of RTK-GPS devices

Let p_i be the local coordinates of each GPS receiver for $i = 1, 2, 3$, as already employed in Eq. (10). Since we already know the optimal inter-GPS distances \mathbf{D} (see Table 3), the optimal set of locations is that one fulfilling

$$\{p_i^*\} = \arg \min_{\{p_i\}} \sum_{(i,j)} (|p_i - p_j| - d_{ij})^2 \quad (11)$$

Before solving the above least square problem, additional constraints must be set since the system is underdetermined. We arbitrarily set the position of the

rear device (index $i = 1$) to be at the origin of the XY plane, that is:

$$p_1 = \begin{bmatrix} 0 \\ 0 \\ 0.132 \end{bmatrix} \quad (12)$$

with the z coordinate conveniently chosen such as the XY plane coincides with the top structure of our vehicle (see Figure 1). There is still an additional degree-of-freedom in the form of a radial symmetry around the Z axis. It can be avoided by arbitrarily imposing that the x axis must exactly bisect the two front GPS devices (with indexes $i = 2$ and $i = 3$, respectively), which agrees with our coordinate system with the x axis pointing forward, as seen in Figure 1(b).

Under these constraints the optimization can be successfully solved, obtaining the sensor coordinates shown in Table 1.

5.2 DGPS characterization

In this section we provide an empirically estimated bound for the Deluo DGPS error.

This is accomplished by batch processing the positioning measurements collected in all the six presented datasets and measuring the root mean square error (RMSE) between the DGPS readings and our ground truth estimation in the time steps where both are available and also synchronized. In the comparison, we considered data from both standalone and DGPS operation modes for the Deluo sensor.

The results yield quite similar RMSE values for the vertical (z coordinate) and horizontal (a combination of x and y coordinates) errors: 13.729 m and 13.762 m, respectively. These figures can be put in contrast with our bounded uncertainty of the ground truth estimate, in the order of ~ 1 cm, thus clearly illustrating the different performance between DGPS and RTK devices.

5.3 Positioning of vertical laser scanners

We have devised an algorithm to automatically calibrate the location of the two vertical SICK LMS-221 laser scanners. To illustrate the underlying idea, please consider the example in Figure 15(a), where the ground truth of the vehicle path (the continuous line) and a first rough estimate of the left vertical laser location on the vehicle are enough to build the 3-d point cloud also shown in the figure. Our algorithm relies on the realization that the *optimal* sensor localization is that one maximizing the matching between the point clouds generated each time the vehicle passes by the same area.

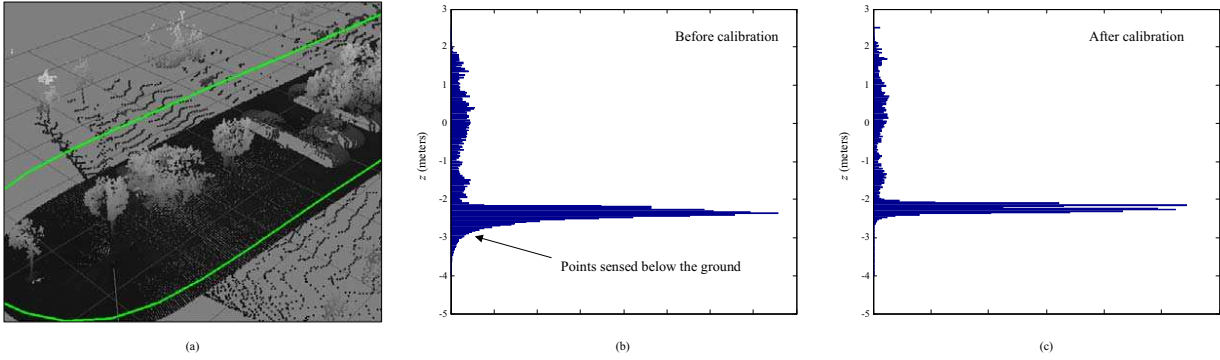


Fig. 15 (a) An example of the kind of 3-d point clouds used to calibrate the onboard location of vertical laser scanners. (b) For the first manually measured location, clear slope errors can be observed in the reconstructed ground, as indicated by the histogram with several values up to 1 meter below the actual ground plane (around $-2.2m$). (c) After calibrating the sensor location, the reconstructed 3-d points are more consistent with the real scenario.

For instance, it can be seen how in Figure 15(a) the vehicle first scans the central area while moving in one direction, then turns and scans it again from the opposite direction.

By identifying a dozen of segments from all the datasets fulfilling this requisite, a Levenberg-Marquardt optimization has been applied to each laser scanner giving us the poses shown in Table 1. The optimized values only differ from those measured manually by a few centimeters and less than 7 degrees in the angles.

To quantify the improvement due to the optimized sensor poses, we have computed the histograms of the z coordinate in two point clouds, both with manually measured and optimized coordinates, in Figure 15(b)–(c), respectively. Given that the coordinate system reference is on the top structure of our vehicle (see Figure 1) and its approximate height is 2.2m to the ground, a distinctive peak can be expected at -2.2m in the histograms due to large portions of the scans being points on the ground. This peak is clearly visible in the graphs, and the interesting observation is that the peak is narrower after calibrating the sensor, which is mainly due to a correction in the angular offsets in the sensor location.

5.4 Cameras

In general, camera calibration involves estimating a set of parameters regarding its projective properties and its pose within a global reference system.

The projective properties of the camera are defined by both its intrinsic and distortion parameters, which determine the position where a certain 3-d point in the scene is projected onto the image plane. Without considering any lens distortion phenomena, a 3-d point $p = (X, Y, Z)^T$ (related to the camera reference system)

projects to an image point $p' = (uw, vw, w)^T$, with w being a scale factor, through the pin-hole model equation:

$$\begin{pmatrix} uw \\ vw \\ w \end{pmatrix} = \mathbf{R} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (13)$$

where f_x and f_y stand for the focal length in units of pixel width and height, respectively, (c_x, c_y) are the coordinates of the principal point of the camera (also known as the image center), and (u, v) represent the projected point coordinates within the image reference system, in pixel units.

Besides, cameras are typically affected by distortion, which produces a displacement of the projected image points and whose major components are known as ra-

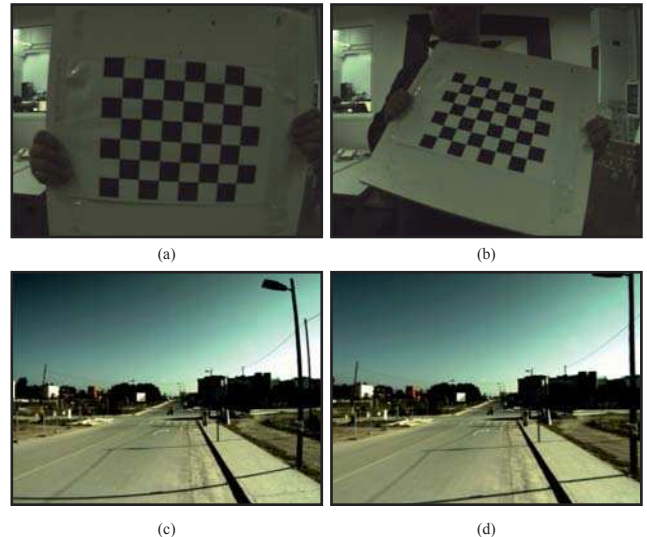


Fig. 16 Camera calibration. (a-b) Two examples of the images employed to estimate both the intrinsic and distortion parameters. (c-d) Raw and rectified versions of one of the images in the CAMPUS-0L dataset.

	Left Camera	Right Camera
f_x (px)	923.5295	911.3657
f_y (px)	922.2418	909.3910
c_x (px)	507.2222	519.3951
c_y (px)	383.5822	409.0285
k_1	-0.353754	-0.339539
k_2	0.162014	0.140431
p_1	$1.643379 \cdot 10^{-03}$	$2.969822 \cdot 10^{-04}$
p_2	$3.655471 \cdot 10^{-04}$	$-1.405876 \cdot 10^{-04}$

Table 4 Calibrated intrinsic parameters of the cameras.

dial and tangential distortion. Thus, the pin-hole projection model can be extended as:

$$u' = u + u^r + u^t \quad (14)$$

$$v' = v + v^r + v^t \quad (15)$$

being the radial terms

$$u^r = u (k_1 r^2 + k_2 r^4) \quad (16)$$

$$v^r = v (k_1 r^2 + k_2 r^4) \quad (17)$$

and the tangential ones

$$u^t = 2p_1 uv + p_2 (r^2 + 2u^2) \quad (18)$$

$$v^t = p_1 (r^2 + 2v^2) + 2p_2 uv \quad (19)$$

with $r = \sqrt{u^2 + v^2}$.

In this work, the intrinsic and distortion parameters of the cameras are estimated through the Zhang's well-known procedure presented in [24]. As the input for the calibration method we employed a sequence of images showing a calibration pattern (in concrete, a checker-board), from several different points of view – see Figure 16(a-b) for an example of the calibration images. Figures 16(c)-(d) show the raw and rectified versions of a representative image from the CAMPUS-0L dataset, respectively. The so-obtained intrinsic parameters are shown in table 4.

On the other hand, an initial rough estimation of the camera pose on the vehicle was obtained by means of a measuring tape. Subsequently, it was refined by applying a Levenberg-Marquardt optimization for a set of points with known 3-d coordinates (from point clouds generated by the vertical scanners) and manually selected pixel coordinates. The algorithm minimizes the overall square error of the point projections in the images, resulting in the final camera poses that can be seen in Table 1.

5.5 IMU

As mentioned above, the MTi sensor position on the vehicle is not relevant because it only measures the attitude and heading of the vehicle. However, it may be

interesting to roughly determine its performance during the vehicle navigation, which can be accomplished by comparing the MTi readings with the estimated angles provided by the ground truth. Since the vehicle movement through the environment may be considered locally almost-planar, in the comparison we focus on the *yaw* angle (i.e. the vehicle heading), as it undergoes larger variations than the attitude components. For this comparison experiment, we use the data from the CAMPUS-0L dataset because the ground truth estimation is available in almost the whole trajectory.

Figure 17(a) shows a comparison between the *yaw* angle provided by the IMU and that computed from the ground truth. The vehicle trajectory is shown in Figure 10(b), where it can be seen a complete 180 deg. turn performed at the end of the campus boulevard. This is illustrated in Figure 17(a) around $t = 250s$, where it can be noticed that the IMU readings indicate a turn of about 200 deg., which represents a severe drift with respect to the ground truth.

This drift becomes more evident in the peaks present in Figures 17(b)-(c), where they are shown both the *yaw* velocity (i.e. its derivative) and the root square error of the IMU *yaw* velocity with respect to the ground truth at each time step. This device performance seems to be acceptable in most of the navigation but its data can not be considered reliable when experiencing noticeable turns. The RMSE value of this error is about 1.631 deg/s, being the mean and the maximum velocities 1.199 deg/s and 14.354 deg/s, respectively.

For the measured *pitch* velocity, we obtain a RMSE of 0.812 deg/s, while 0.092 deg/s and 0.487 deg/s are the values of the mean and the maximum velocities, respectively. Finally, values for the RMSE, mean and maximum *roll* velocities are 0.609 deg/s, 0.149 deg/s, and 0.882 deg/s, respectively.

6 Uncertainty Analysis

Comparing the estimate from a given SLAM method against our ground truth can only be significant if uncertainties (both in the SLAM output and the ground truth) are carefully taken into account. In this section we derive an estimate of the probability distribution of our reconstructed vehicle poses v_t . For that goal it is first needed the distribution of the errors in the 3-d locations measured by the three GPS receivers.

6.1 Estimation of RTK-GPS noise

From the sensor characterization, the working principle of RTK-GPS and the manufacturer specifications,

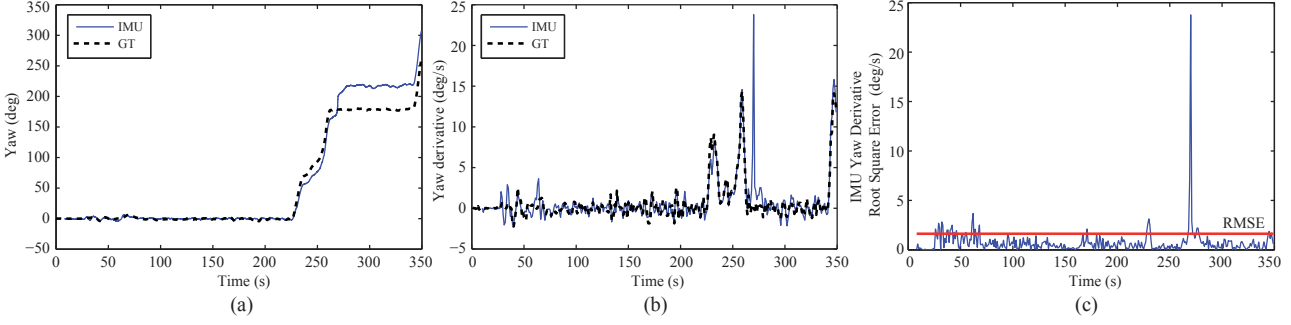


Fig. 17 (a) *Yaw* estimation provided by the MTi device (*IMU*) and the ground truth (*GT*). (b) *Yaw* velocity computed from both data. (c) *IMU* error in *yaw* derivative estimation with respect to the ground truth and corresponding root mean-square-error (RMSE) (Data corresponding to the CAMPUS-0L dataset).

it comes out that noise in latitude and longitude coordinates is smaller than in the height. There is no reason for the statistical errors in the latitude to be different to those in the longitude, and experimental data confirm this point. This characteristic and the fact that the *XY* plane in our transformation to local coordinates is tangent to the GPS ellipsoid imply that the measuring errors in *x* and *y* should be equal. Let denote the standard deviation of these errors as σ_x and σ_y , respectively, being the error in the height (*z* axis) modeled by σ_z .

A characteristic of RTK-GPS technology is that errors in *z* are always larger than those in (*x*,*y*). We model this fact by introducing a scale factor η such as

$$\begin{aligned}\sigma_x &= \sigma_y = \sigma_0 \\ \sigma_z &= \eta\sigma_0\end{aligned}\quad (20)$$

By grabbing thousands of GPS readings with the vehicle stopped, we have experimentally determined this factor to be approximately $\eta = 1.75$.

At this point we only need to estimate one parameter (σ_0) to have a complete model of the GPS errors. Our approach to estimate σ_0 relies on the information given by the residuals of the optimization of \mathbf{D} (the inter-GPS distances), derived in Section 4.2. Since we know that those distances must be constant, the probability distribution of the deviations with respect to the optimal value, shown in Figure 18 as histograms, can be used to characterize the GPS localization errors.

In particular, we will focus on the probability density distribution (pdf) of d_{ij}^2 , being d_{ij} the distance between two 3-d points measured by the GPS receivers *i* and *j*. It can be proven that this density, which cannot be approximated by first order uncertainty propagation, is a function of the error in *x* and *y* (σ_0) and the relative locations of the GPS receivers on the vehicle

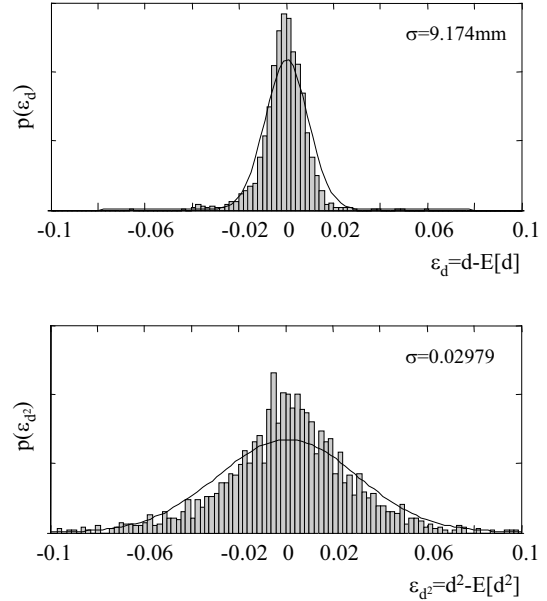


Fig. 18 Histograms of (top) the residuals from the optimization of d_{ij} in Section 4.2, and (bottom) the same for the square distances. Units are meters and square meters, respectively. The variance of the square distance residuals is a convenient value for modeling the 3-d positioning errors of the GPS receivers.

l_{ij} , as estimated in Section 5.1. Thus we can express its variance as:

$$\sigma_{d^2}^2 = f(\sigma_0, \{l_{ij}\}) \quad (21)$$

Since the values of l_{ij} are known, we can accurately estimate σ_0 from Eq. (21) through a Monte-Carlo (MC) simulation, arriving to the numerical results

$$\begin{aligned}\sigma_x &= \sigma_y = 0.64cm \\ \sigma_z &= 1.02cm\end{aligned}\quad (22)$$

which characterize the errors of the GPS 3-d measurements. These values predict a standard deviation in the

inter-GPS distances of 9.1mm , which is consistent with the residuals of the LM optimization in Table 3.

Next sections rely on this error model to finally provide the uncertainty of the vehicle path.

6.2 Uncertainty in the vehicle 6-d pose

Our model for the pose uncertainty is a multivariate Gaussian distribution, such as its mean \hat{v}_t coincides with the optimal value derived in Eq. (10) and its 6×6 covariance matrix W_t , is determined next. That is,

$$v_t \sim \mathcal{N}(\hat{v}_t, W_t)$$

The covariance matrix W_t depends on the vehicle orientation, thereby varying with time. Furthermore, it should be estimated in each case from a Monte-Carlo (MC) simulation which provides a better approximation than linearization approaches. To avoid executing one MC estimation for every possible v_t we propose to only estimate the covariance for the case of the vehicle at the origin, then rotate this reference covariance according to the real attitude angles of the vehicle. That is, if W^* stands for the reference covariance matrix,

$$W_t = J(v_t)W^*J(v_t)^\top \quad (24)$$

where the Jacobian of the transformation $J(v_t)$ incorporates the 3×3 rotation matrix $R(v_t)$ associated to the vehicle pose v_t :

$$J(v_t) = \begin{bmatrix} R(v_t) & 0 \\ 0 & I_3 \end{bmatrix} \quad (25)$$

By means of 10^7 simulations of noisy GPS measurements we arrive to the reference matrix shown in Eq. (23) and pictured in Figure 19 as 95% confidence ellipsoids. This covariance matrix is specific for the noise levels of our GPS receivers and their relative positions on the vehicle, thus it would not be applicable to other configurations.

Our public datasets available for download incorporate the precomputed values of the covariance matrix rotated accordingly to Eq. (24) for each timestep.

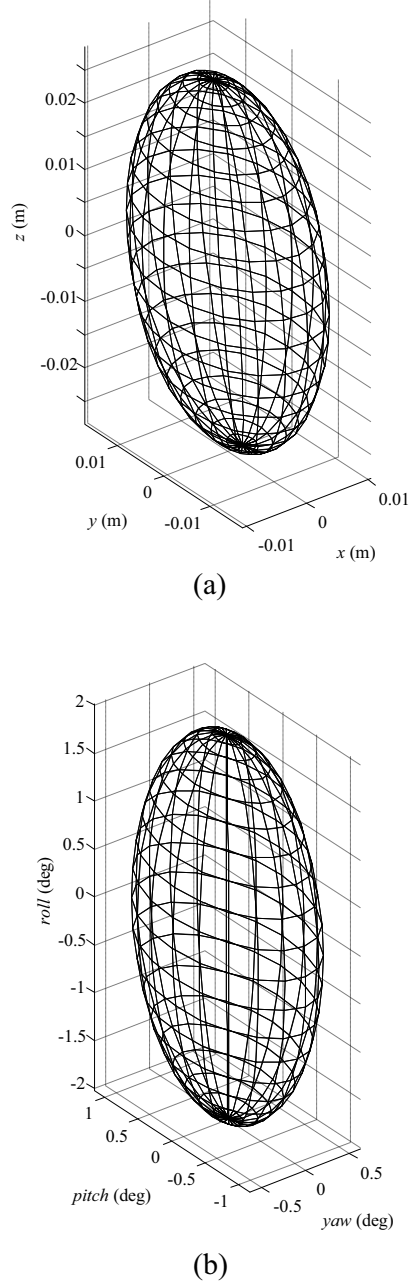


Fig. 19 The 95% confidence ellipsoids for the uncertainty of the vehicle 6-d pose, represented separately for (x, y, z) (a) and the three attitude angles (b).

$$W^* = \begin{bmatrix} 1.4472 \cdot 10^{-5} & -7.6603 \cdot 10^{-9} & -7.5266 \cdot 10^{-6} & 1.2761 \cdot 10^{-8} & -6.5127 \cdot 10^{-6} & 3.4478 \cdot 10^{-8} \\ -7.6603 \cdot 10^{-9} & 3.6903 \cdot 10^{-5} & 1.3863 \cdot 10^{-7} & -1.7907 \cdot 10^{-5} & 1.1747 \cdot 10^{-7} & 2.0664 \cdot 10^{-5} \\ -7.5266 \cdot 10^{-6} & 1.3863 \cdot 10^{-7} & 1.0407 \cdot 10^{-4} & -1.5037 \cdot 10^{-7} & 6.0132 \cdot 10^{-5} & -1.9469 \cdot 10^{-7} \\ 1.2761 \cdot 10^{-8} & -1.7907 \cdot 10^{-5} & -1.5037 \cdot 10^{-7} & 1.5434 \cdot 10^{-5} & -1.2943 \cdot 10^{-7} & -7.3799 \cdot 10^{-7} \\ -6.5127 \cdot 10^{-6} & 1.1747 \cdot 10^{-7} & 6.0132 \cdot 10^{-5} & -1.2943 \cdot 10^{-7} & 5.2093 \cdot 10^{-5} & -1.2652 \cdot 10^{-7} \\ 3.4478 \cdot 10^{-8} & 2.0664 \cdot 10^{-5} & -1.9469 \cdot 10^{-7} & -7.3799 \cdot 10^{-7} & -1.2652 \cdot 10^{-7} & 1.5840 \cdot 10^{-4} \end{bmatrix} \quad (23)$$

6.3 Trajectory of the sensors

Many existing SLAM techniques assume that the origin of the *robocentric* coordinate system coincides with that of the sensor – MonoSLAM, or SLAM based on monocular cameras, is a prominent example [7]. This approach is clearly convenient when there is only one sensor (e.g. one camera in MonoSLAM or one laser scanner in 2-d SLAM).

A ground truth for those methods can be easily obtained from the vehicle trajectory v_t derived in Section 4.3 and the local coordinates of the sensor of interest, denoted by u in the following. A summary of these local coordinates for our vehicle is presented in Table 1.

For each timestep t , the pose of the given sensor in global coordinates s_t is obtained as the result of the 6-d pose composition

$$s_t = v_t \oplus u \quad (26)$$

Taking into account that both v_t and u have associated uncertainties, modeled by the covariance matrices W_t and U , we also represent the global coordinates as multivariate Gaussians with mean \hat{s}_t and covariance S_t . The mean is simply given by Eq. (26) applied to the means of the vehicle path and the local coordinates, while the derivation of the covariance is detailed in Appendix A.

For the purpose of estimating the ground truth for a sensor path s_t , we could assume a perfect knowledge of the local coordinates, which implies a null U matrix. However, for the sake of accuracy this matrix should roughly model the potential errors in the process of manually measuring these quantities (or in the optimization methods used for some sensors, e.g. the cameras in Section 5.4). The sensor paths in the published datasets assume a standard deviation of 0.02 mm and 0.1 degrees for all the translations and rotations, respectively.

6.4 A measure of the ground truth consistency

Each pose in the ground truth of the vehicle trajectory can be seen as our estimate of the corresponding *random variable*. As such, we can only expect to obtain an optimal estimation and its corresponding uncertainty bound, as derived in previous sections.

However, in our problem there is an additional constrain which can be used to provide a measure of the consistency for each ground truth point v_t . The constrain is the rigid body assumption, and as discussed next will determine how much confident shall we be

about the estimate of each vehicle pose. It must be highlighted that this measure cannot fix in any way the optimal estimations already derived in previous sections.

Our derivation starts with the vector \mathbf{D} of square distances between each different pair of GPS sensors, that is:

$$\mathbf{D} = [d_{12}^2 \ d_{13}^2 \ d_{23}^2] = f(\{x_i, y_i, z_i\}) \quad , \quad i = 1, 2, 3$$

Since each distance is a function of the measured 3-d locations (x_i, y_i, z_i) ,

$$d_{ij}^2 = (x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 \quad (27)$$

the covariance matrix of D can be estimated by linearization as follows:

$$\begin{aligned} \Sigma_{\mathbf{D}} &= \mathbf{F} \text{diag}(\sigma_x^2, \sigma_x^2, \sigma_x^2, \sigma_y^2, \sigma_y^2, \sigma_y^2, \sigma_z^2, \sigma_z^2, \sigma_z^2) \mathbf{F}^\top \\ &= \sigma_0^2 \mathbf{F} \text{diag}(1, 1, 1, 1, 1, 1, \eta^2, \eta^2, \eta^2) \mathbf{F}^\top \end{aligned}$$

where in the last step we employed the definitions in Eq. (20).

Let $\Delta_{d_{ij}}$ be the distance between the points i and j in the XY plane, and $\Delta_{z_{ij}} = z_i - z_j$ their distance in z . Since the GPS units are roughly at the same height in our vehicle, it turns out that $\Delta_{z_{ij}}^2 \ll \Delta_{d_{ij}}^2$. By using this approximation, it can be shown after some operations that the covariance matrix $\Sigma_{\mathbf{D}}$ becomes:

$$4\sigma_0^2 \begin{bmatrix} 2\Delta_{d_{12}}^2 & \Delta_{12}\Delta_{13}\cos\phi_{13}^{12} - \Delta_{12}\Delta_{23}\cos\phi_{23}^{12} \\ \Delta_{12}\Delta_{13}\cos\phi_{13}^{12} & 2\Delta_{d_{13}}^2 & \Delta_{13}\Delta_{23}\cos\phi_{23}^{13} \\ -\Delta_{12}\Delta_{23}\cos\phi_{23}^{12} & \Delta_{13}\Delta_{23}\cos\phi_{23}^{13} & 2\Delta_{d_{23}}^2 \end{bmatrix}$$

Here it has been employed ϕ_{ik}^{ij} as the angle between the pair of 2-d vectors passing by the GPS sensors i and j , and i and k , respectively. Disregarding small variations due to terrain slopes, these angles and all the distances in Eq. (28) remain constant with time, thus it provides a good approximation of the covariance of \mathbf{D} during all the datasets.

We can now define our measure of the quality of the GPS positioning at each timestep t as the Mahalanobis distance between the vector of measured square distances \mathbf{D}_t and the optimized values $\hat{\mathbf{D}}$, using the above derived covariance matrix $\Sigma_{\mathbf{D}}$, that is:

$$q_t = \sqrt{(\mathbf{D}_t - \hat{\mathbf{D}})^\top \Sigma_{\mathbf{D}}^{-1} (\mathbf{D}_t - \hat{\mathbf{D}})} \quad (28)$$

Lower values of this measure reflect a better quality in the ground truth. We illustrate the behavior of this confidence measure in Figure 20 for three different datasets. It can be remarked how the measure remains relatively constant with time, thus guaranteeing

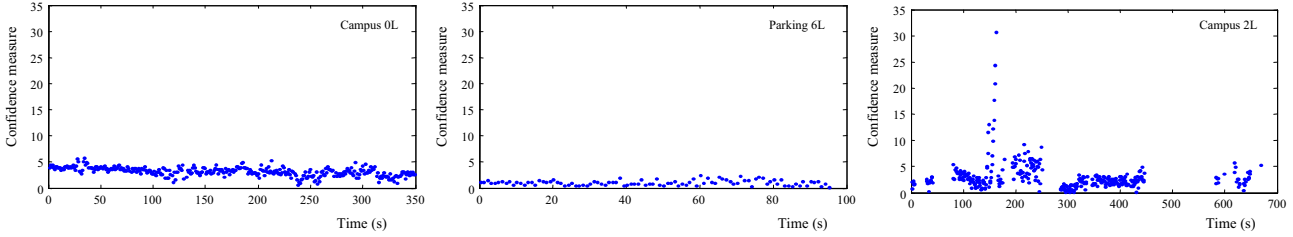


Fig. 20 Our measure of the confidence of the ground truth for three different datasets. Lower values mean that the measured 3-d positions for the GPS devices are closer to the condition of rigid rotations, thus they must be more accurate.

the accuracy of our estimated ground truth. Regarding the high values observed about $t = 170$ for the dataset CAMPUS-2L, they coincide with a moderate slope in the terrain, which may render our approximation of Σ_D less accurate for those specific segments of the dataset. Therefore, our quality measure should be taken more into account when the vehicle moves in relatively more horizontal areas, e.g. all three *parking* datasets.

7 Validation

Quantitative validations have been already presented along the paper. For example, the narrow distribution of the residuals in Figure 18(a) implies that our reconstructed paths have errors in the order of $1cm$. The uncertainties derived in Section 6.2 confirm our expectation of centimeter-level errors.

In addition, this section provides qualitatively illustrations of the accuracy of the reconstructed trajectories for the vehicle and the sensors by means of building different kinds of maps directly from the ground truth.

7.1 Construction of 3-d point clouds

To validate the accuracy of the path ground truth and the calibration of vertical laser scanners, we can compare, by visual inspection, the actual shape of a representative real object in the scene with its 3-d cloud point structure generated from laser data and the vehicle path.

In that sense, consider Figure 9(e), corresponding to PARKING-6L, which shows a group of cars parked under some trees. Please note how some of the cars had at least two of their sides scanned by the vertical laser scanners, as the vehicle navigated around them in different directions. It can be appreciated in the figure that the generated 3-d point cloud describing the cars visually fits their real shape quite accurately, revealing a considerable consistency between the real world and

its corresponding point cloud representation¹². Therefore, both ground truth estimation and laser scanners calibration can be considered as notably reliable. Nevertheless, it must be remarked that expected errors in the point clouds are larger than those in the vehicle path, since the projected points magnify the effects of small orientation errors in the vehicle.

7.2 Construction of colored 3-d maps

The camera calibration, which involved the estimation of both distortion and projection parameters and its location on the vehicle, can be also validated by determining the real color of the points in the above-mentioned 3-d point clouds.

For that purpose, we re-projected those 3-d points into the captured images, at each time step, and determined their real colors from the re-projected pixel coordinates in the images. As shown in Figure 21(a)–(b) for the CAMPUS-0L dataset, the color information and the 3-d structure of the scene seems to be sensibly coherent. A pair of representative examples can be found in the buildings surrounding the road, whose colors are that of the red bricks which compose their fronts, and in the crosswalk shown in the middle of Figure 21(a), which by visual inspection seems to be properly projected on the floor.

8 Conclusion

In this work we have identified an important unsatisfied need within the SLAM community, namely the existence of a freely accessible compilation of datasets with an associated ground truth suitable for evaluating the different SLAM techniques.

The present work intends to provide such an accurate ground truth for a series of outdoor datasets, including its estimated uncertainty bounds, in the range

¹² The complete 3-d point clouds are published along the dataset files.

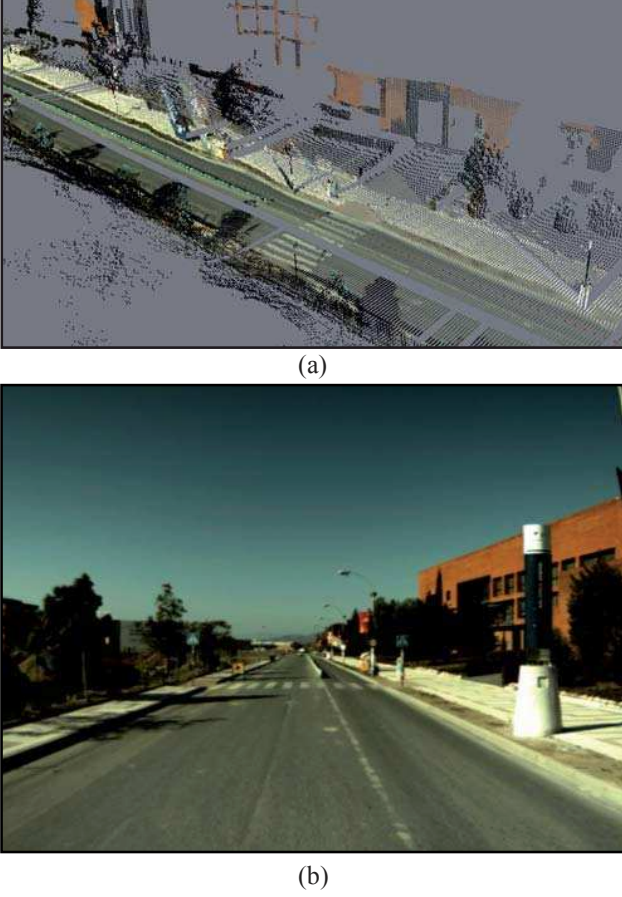


Fig. 21 (a) A snapshot of the colored 3-d point map employed for the validation of cameras calibration and (b) its the corresponding image from the CAMPUS-0L dataset.

of 1cm and 0.5deg. These bounds remain constant with time along all the datasets, turning the datasets into an ideal testbed for the evaluation of methods such as visual SLAM or visual odometry. The scalability of SLAM methods can be also assessed with regard to their capability of closing loops, since the more than 6Km of recorded datasets comprise several loops of different lengths.

As an additional result of our ground truth estimation, we also provide reference 3-d point clouds for each of the datasets, which may be proper to test a variety of other techniques such as 3-d surface reconstruction or path planning.

All the datasets are freely available online among with extensive documentation and open-source software which allows the easy and configurable post-processing of the raw data.

Acknowledgements The authors want to thank Topcon for their collaboration in the work related to the RTK GPS receivers.

A Uncertainty propagation with 6-d poses

We are interested in the operation of pose composition (denoted by the operator \oplus), where a given pose b is transformed into the coordinate system described by another pose a , giving as result the new pose c :

$$c = f(a, b) = a \oplus b \quad (29)$$

The 6-d components of c are $(x_c \ y_c \ z_c \ \phi_c \ \chi_c \ \psi_c)^\top$, and similarly for a and b with the appropriate subscripts. The variables ϕ , χ and ψ stand for the three orientation angles *yaw*, *pitch* and *roll*, respectively.

Using homogeneous coordinates it can be shown that the function $f(\cdot)$ becomes:

$$\begin{cases} x_c = x_a + R_{11}x_b + R_{12}y_b + R_{13}z_b \\ y_c = y_a + R_{21}x_b + R_{22}y_b + R_{23}z_b \\ z_c = z_a + R_{31}x_b + R_{32}y_b + R_{33}z_b \\ \phi_c = \phi_a + \phi_b \\ \chi_c = \chi_a + \chi_b \\ \psi_c = \psi_a + \psi_b \end{cases} \quad (30)$$

where it has been used R_{ij} to represent the entries of the rotation matrix associated to the pose a .

We next take into account the uncertainties in each variable by means of linearization of the error propagation. Each pose can be modeled as a multivariate Gaussian distribution, such as:

$$a \sim \mathcal{N}(\hat{a}, A) \quad (31)$$

$$b \sim \mathcal{N}(\hat{b}, B)$$

$$c \sim \mathcal{N}(\hat{c}, C)$$

The mean is given by the composition of the means of the inputs, that is $\hat{c} = \hat{a} \oplus \hat{b}$. Assuming that errors in a and b are independent, the covariance C can be approximated by:

$$C = \frac{\partial f}{\partial a} A \frac{\partial f}{\partial a}^\top + \frac{\partial f}{\partial b} B \frac{\partial f}{\partial b}^\top \quad (32)$$

In this expression, $\frac{\partial f}{\partial a}$ and $\frac{\partial f}{\partial b}$ stand for the Jacobian matrices of the function f with respect to the elements of a and b , respectively. It can be shown that the first Jacobian becomes:

$$\frac{\partial f}{\partial a} = \begin{pmatrix} 1 & 0 & 0 & F_{14} & F_{15} & F_{16} \\ 0 & 1 & 0 & F_{24} & F_{25} & F_{26} \\ 0 & 0 & 1 & 0 & F_{35} & F_{36} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (33)$$

with:

$$\begin{aligned} F_{14} &= -x_b \sin \phi_a \cos \phi_a + \\ &\quad y_b (-\sin \phi_a \sin \chi_a \sin \psi_a - \cos \phi_a \cos \psi_a) + \\ &\quad z_b (-\sin \phi_a \sin \chi_a \cos \psi_a + \cos \phi_a \sin \psi_a) \end{aligned}$$

$$\begin{aligned} F_{15} &= -x_b \cos \phi_a \sin \chi_a + \\ &\quad y_b \cos \phi_a \cos \chi_a \sin \psi_a + z_b \cos \phi_a \cos \chi_a \cos \psi_a \end{aligned}$$

$$\begin{aligned} F_{16} &= y_b (\cos \phi_a \sin \chi_a \cos \psi_a + \sin \phi_a \sin \psi_a) + \\ &\quad z_b (-\cos \phi_a \sin \chi_a \sin \psi_a + \sin \phi_a \cos \psi_a) \end{aligned}$$

$$\begin{aligned}
F_{24} &= x_b \cos \phi_a \cos \chi_a + \\
&\quad y_b (\cos \phi_a \sin \chi_a \sin \psi_a - \sin \phi_a \cos \psi_a) + \\
&\quad z_b (\cos \phi_a \sin \chi_a \cos \psi_a + \sin \phi_a \sin \psi_a) \\
F_{25} &= -x_b \sin \phi_a \sin \chi_a + y_b \sin \phi_a \cos \chi_a \sin \psi_a + \\
&\quad z_b \sin \phi_a \cos \chi_a \cos \psi_a \\
F_{26} &= y_b (\sin \phi_a \sin \chi_a \cos \psi_a - \cos \phi_a \sin \psi_a) + \\
&\quad z_b (-\sin \phi_a \sin \chi_a \sin \psi_a - \cos \phi_a \cos \psi_a) \\
F_{35} &= -x_b \cos \chi_a - y_b \sin \chi_a \sin \psi_a - z_b \sin \chi_a \cos \psi_a \\
F_{36} &= y_b \cos \chi_a \cos \psi_a - z_b \cos \chi_a \sin \psi_a
\end{aligned}$$

Regarding the Jacobian with respect to b , it is given by a block diagonal matrix composed of the rotation matrix R (for the angles of pose a) and the identity I :

$$\frac{\partial f}{\partial b} = \begin{bmatrix} R(a) & 0 \\ 0 & I_3 \end{bmatrix}. \quad (34)$$

References

1. P.J. Besl and N.D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
2. J.L. Blanco. The Mobile Robot Programming Toolkit (MRPT) website, 2008.
3. A. Bonarini, D. Migliore, G. Fontana, and M. Matteucci. The Raw Seeds project website, 2009.
4. K.M. Borkowski. Transformation of geocentric to geodetic coordinates without approximations. *Astrophysics and Space Science*, 139(1):1–4, 1987.
5. J. Campbell, R. Sukthankar, I. Nourbakhsh, and A. Pahwa. A Robust Visual Odometry and Precipice Detection System Using Consumer-grade Monocular Vision. In *IEEE International Conference on Robotics and Automation*, pages 3421–3427, 2005.
6. J. Civera, A.J. Davison, and JMM Montiel. Inverse depth parametrization for monocular SLAM. *IEEE Trans. Robotics*, 24(5), 2008.
7. A.J. Davison, I. Reid, N. Molton, and O. Stasse. MonoSLAM: Real-Time Single Camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6), June 2007.
8. C. De Boor. *A Practical Guide to Splines*. Springer, 2001.
9. SP Drake. Converting GPS coordinates $\lambda \varphi h$ to local coordinates enu, 2000.
10. H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part I. *IEEE Robotics & Automation Magazine*, 13(2):99–110, 2006.
11. J. Guivant, E. Nebot, J. Nieto, and F. Masson. Navigation and mapping in large unstructured environments. *International Journal of Robotics Research*, 23(4):449–472, 2004.
12. J.E. Guivant and E.M. Nebot. Optimization of the simultaneous localization and map-building algorithm for real-time implementation. *IEEE Transactions on Robotics and Automation*, 17(3):242–257, 2001.
13. B.K.P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987.
14. A. Howard and N. Roy. The robotics data set repository (radish), 2003.
15. M. Kaess, A. Ranganathan, and F. Dellaert. Fast incremental square root information smoothing. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, pages 2129–2134, 2007.
16. A. Leick. *GPS Satellite Surveying*. Wiley, 2004.
17. D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM J. Appl. Math.*, 11(2):431–441, 1963.
18. M. Montemerlo. *FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem With Unknown Data Association*. PhD thesis, University of Washington, 2003.
19. J.J. Mor. The Levenberg-Marquardt algorithm: implementation and theory. *Lecture Notes in Mathematics*, 630:105–116, 1977.
20. J. Nieto, J. Guivant, E. Nebot, and S. Thrun. Real time data association for FastSLAM. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, 2003.
21. Lina M. Paz, J. Guivant, J. D. Tardós, and J. Neira. Data association in $O(n)$ for divide and conquer SLAM. In *Robotics: Science and Systems, RSS*, Atlanta, GA, USA, June 2007.
22. R. Smith, M. Self, and P. Cheeseman. A stochastic map for uncertain spatial relationships. *The fourth international symposium on Robotics Research*, pages 467–474, 1988.
23. M. Walter, R. Eustice, and J. Leonard. A provably consistent method for imposing sparsity in feature-based SLAM information filters. In *International Symposium of Robotics Research (ISRR)*. Springer, 2004.
24. Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11):1330–1334, 2000.