DOI 10.1007/s11263-011-0441-3

1-Point-RANSAC Structure from Motion for Vehicle-Mounted Cameras by Exploiting Non-holonomic Constraints

Davide Scaramuzza

Received: 6 May 2009 / Accepted: 25 March 2011 / Published online: 7 April 2011

Abstract This paper presents a new method to estimate the relative motion of a vehicle from images of a single camera. The computational cost of the algorithm is limited only by the feature extraction and matching process, as the outlier removal and the motion estimation steps take less than a fraction of millisecond with a normal laptop computer. The biggest problem in visual motion estimation is data association; matched points contain many outliers that must be detected and removed for the motion to be accurately estimated. In the last few years, a very established method for removing outliers has been the "5-point RANSAC" algorithm which needs a minimum of 5 point correspondences to estimate the model hypotheses. Because of this, however, it can require up to several hundreds of iterations to find a set of points free of outliers. In this paper, we show that by exploiting the nonholonomic constraints of wheeled vehicles it is possible to use a restrictive motion model which allows us to parameterize the motion with only 1 point correspondence. Using a single feature correspondence for motion estimation is the lowest model parameterization possible and results in the two most efficient algorithms for removing outliers: 1-point RANSAC and histogram voting. To support our method we run many experiments on both synthetic and real data and compare the performance with a state-of-the-art approach. Finally, we show an application of our method to visual odometry by recovering a 3 Km trajectory in a cluttered urban environment and in real-time.

Keywords Outlier removal · Ransac · Structure from motion

1 Introduction

Vehicle ego-motion estimation is a key component for autonomous driving and computer vision based driving assistance. Using cameras instead of other sensors for computing ego-motion allows a simple integration of egomotion data into other vision based algorithms, such as obstacle, pedestrian, and lane detection, without the need for calibration between sensors. This reduces maintenance and cost. While there exist nowadays a wide availability of algorithms for motion estimation using video input alone (see Sect. 2), cameras are still little integrated in the motion estimation system of a mobile robot and even less in that of an automotive vehicle. The main reasons for this are the following:

- many algorithms assume static scenes and cannot cope with dynamic and cluttered environments or huge occlusions by other passing vehicles
- the data-association problem (feature matching and outlier removal) is not completely robust and can fail,
- the motion estimation scheme usually requires many keypoints and can fail when only a few keypoints are available in almost absence of structure.

In this paper, we show that all these areas can be improved by using a restrictive motion model which allows us to parameterize the motion with only 1 point correspondence. The first consequence is that only one feature correspondence suffices for computing the epipolar geometry. This allows motion to be estimated also in those cases where there is only a few number of features available and hence standard algorithms would fail. The most valuable consequence

Please observe that this paper is accompanied by a demonstrative video available at: http://www.youtube.com/watch?v=t7uKWZtUjCE.

D. Scaramuzza (✉)
Autonomous Systems Lab, ETH Zurich, Zurich, Switzerland
e-mail: davide.scaramuzza@ieee.org

is that very efficient methods for removing outliers can be implemented. Once the outliers are removed, the motion can be refined using all the inliers.

The structure of the paper is the following. In Sect. 2, we review the related work. In Sect. 3, we give a short description of the RANSAC paradigm. In Sect. 4, we explain how the nonholomic constraints of wheeled vehicles allow us to parameterize the motion with a single point correspondence. In Sect. 5, we describe two efficient methods for removing the outliers by taking advantage of the proposed motion model. Finally, in Sects. 6 and 7 we present our experimental results and conclusions.

2 Related Work on Visual Motion Estimation

Most of the works in estimating vehicle motion using vision (also called visual odometry) has been produced using stereo cameras (Moravec 1980; Lacroix et al. 1999; Jung and Lacroix 2005; Nister et al. 2006; Maimone et al. 2007). Nevertheless, visual odometry methods for outdoor applications have also been produced, which use a single camera alone. The problem of recovering relative camera poses and 3D structure from a set of monocular images has been largely studied for many years and is known in the computer vision community as “Structure From Motion” (SFM) (Hartley and Zisserman 2004). Successful results with only a single camera and over long distances (from hundreds of meters up to kilometers) have been obtained in the last decade using both perspective and omnidirectional cameras (see Nister et al. 2006; Corke et al. 2004; Lhuillier 2005; Goecke et al. 2007; Tardif et al. 2008; Milford and Wyeth 2008; Scaramuzza and Siegwart 2008). Here, we review some of these works.

Related works can be divided into three categories: feature-based methods, appearance based methods, and hybrid methods. Feature-based methods are based on salient and repetitive features that are tracked over the frames; appearance based methods use the intensity information of all the pixels in the image or of subregions of it; hybrid methods use a combination of the previous two.

In the first category are the works of Nister et al. (2006), Corke et al. (2004), Lhuillier (2005), Tardif et al. (2008). In Nister et al. (2006), Nister et al. dealt with the case of a stereo camera but they also provided a monocular solution implementing a fully structure from motion algorithm that takes advantage of the 5-point algorithm and RANSAC. In Corke et al. (2004), Corke et al. provided two approaches for monocular visual odometry based on omnidirectional imagery from a catadioptric camera. They performed experiments in the desert and therefore used keypoints from the ground plane. In Lhuillier (2005), Lhuillier used 5-point RANSAC and bundle adjustment to recover both the motion and the 3D map. In Tardif et al. (2008), Tardif et al. presented an approach for incremental and accurate SFM from a car over a very long run (2.5 Km) without bundle adjustment. To achieve it, they decoupled the rotation and translation estimation. In particular, they estimated the rotation using points at infinity and the translation from the recovered 3D map. Bad correspondences were removed with preemptive 5-point RANSAC (Nister 2005).

Among the appearance based or hybrid approaches are the works of Goecke et al. (2007), Milford and Wyeth (2008), Scaramuzza and Siegwart (2008). In Goecke et al. (2007), Goecke et al. used the Fourier-Mellin Transform for registering perspective images of the ground plane taken from a car. In Milford and Wyeth (2008), Milford et al. presented a method to extract approximate rotational and translational velocity information from a single perspective camera mounted on a car, which was then used in a RatSLAM scheme (Milford et al. 2004). However, appearance based approaches alone are not very robust to occlusions. For this reason, in our previous works (Scaramuzza and Siegwart 2008; Scaramuzza et al. 2008), we used appearance to estimate the rotation of the car and features from the ground plane to estimate the translation and the absolute scale. The feature-based approach was also used as a firewall to detect failure of the appearance based method.

Closely related to structure from motion is what is known in the robotics community as Simultaneous Localization and Mapping (SLAM), which aims at estimating the motion of the robot while simultaneously building and updating a coherent environment map. In the last years successful results have been obtained also using single cameras (see Deans 2002; Davison 2003; Clemente et al. 2007, and Lemaire and Lacroix 2007).

3 Minimal Model Parameterizations and RANSAC

For unconstrained motion (6DoF) of a calibrated camera the minimum number of point correspondences required for solving the relative pose problem is five (see 5-point algorithm of Nister 2003; Stewenius et al. 2006). This can be intuitively understood by noticing that of the six parameters that we need to estimate (three for the rotation and three for the translation) only five are actually required. Indeed, the relative pose between two cameras is always valid up to a scale.

The first solution to the 5-point relative pose problem was proven by Kruppa in 1913 (Kruppa 1913) to have at most eleven solutions. This was later improved by Faugeras and Maybank (1990) showing that there are at most ten solutions but the method found only in 2003 its efficient implementation in the algorithm of Nister (2003) and Stewenius et al. (2006). Before this efficient version of the 5-point algorithm, the most common methods used to solve the relative pose problem were the 8-point, 7-point, and 6-point algorithms, which are all still widely used. The 8 and 7-point methods relaxed the requirements of having calibrated cameras and hence led very efficient and easy-to-implement algorithms. The 8-point algorithm (Longuet-Higgins 1981) has a linear solver for a unique solution while the 7-point method (Hartley and Zisserman 2004) leads to up to three solutions. The 6-point method (Philip 1996; Pizarro et al. 2003) works for calibrated cameras and yields up to six solutions.

An interesting review and comparison of all these methods can be found in Stewenius et al. (2006). There, it is shown that the new implementation of the 5-point method provides superior pose estimates with respect to all the other algorithms.

3.1 RANSAC

In every situation where a model has to be estimated from given data, we have to deal with outliers. The *random sample consensus* (RANSAC) (Fischler and Bolles 1981) has been established as the standard method for model estimation in the presence of outliers. Structure from motion is one application of the RANSAC scheme. The estimated model is the motion (**R**, **T**) and it is estimated from feature correspondences. Outliers are feature points with wrong data-associations. The idea behind RANSAC is to compute model hypotheses from randomly-sampled minimal sets of data points and then verify these hypotheses on the other data points. The hypothesis that shows the highest consensus with the other data is selected as solution. The number of subsets (iterations) N that is necessary to guarantee that a correct solution is found can be computed by

$$N = \frac{\log(1-p)}{\log(1-(1-\varepsilon)^s)} \tag{1}$$

where s is the number of minimal data points, ε is the percentage of outliers in the data points, and p is the requested probability of success (Fischler and Bolles 1981). N is exponential in the number of data points necessary for estimating the model, so there is a high interest in finding the minimal parameterization of the model. For unconstrained motion (6DoF) of a calibrated camera this would be 5 correspondences. Using the 6, 7, or 8-point method would increase the number of necessary iterations and therefore slow down the motion estimation algorithm. It is therefore of utmost importance to find the minimal parameterization of the model to estimate. In the case of planar motion, the motion-model complexity is reduced (3DoF) and can be parameterized with 2 points as described in Ortin and Montiel (2001).

For wheeled vehicles we will show in Sect. 4 that an even more restrictive motion model can be chosen which allows us to parameterize the motion with only 1 feature correspondence. Using a single feature correspondence for motion estimation is the lowest model parameterization possible and results in the most efficient RANSAC algorithm. We will also show that an even more efficient algorithm can be devised, which requires no iteration.

Table 1

Min. set of points	8	7	6	5	2	1
No. of iterations	1177	587	292	145	16	7

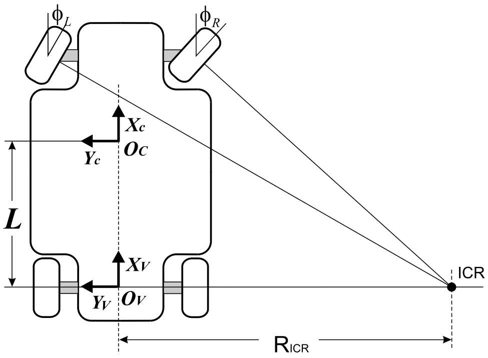


Fig. 1 General Ackermann steering principle

A summary of the number of RANSAC iterations needed as a function of the number of model parameters s is shown in Table 1. These values were obtained assuming a probability of success $p = 99\%$ and a percentage of outliers $\varepsilon = 50\%$.

4 Why Do We Need Only 1 Point?

For a wheeled vehicle to exhibit rolling motion, a point must exist around which each wheel of the vehicle follows a circular course (Siegwart et al. 2011). This point is known as Instantaneous Center of Rotation (ICR) and can be computed by intersecting all the roll axes of the wheels (Fig. 1). This property holds for any robot. In particular for car-like and differential-drive. For cars the existence of the ICR is ensured by the Ackermann steering principle (Siegwart et al. 2011). This principle ensures a smooth movement of the vehicle by applying different steering angles to the inner and outer front wheel while turning (see Fig. 1).

As the reader can perceive, the motion of a camera fixed on the vehicle can then be locally described with circular motion (note, rectilinear motion can be represented along a circle with infinite radius of curvature). This constraint reduces the degrees of freedom of motion to two, namely the rotation angle and the radius of curvature. Therefore, only one feature correspondence suffices for computing the relative pose up to a scale. As we will see in the next section,

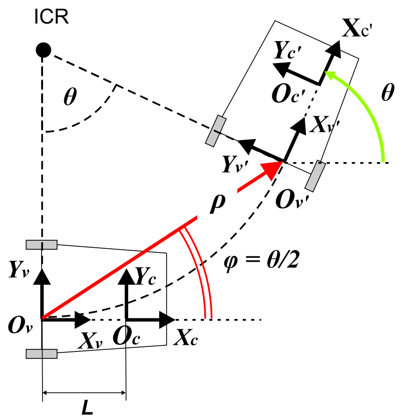


Fig. 2 Relation between camera axes in circular motion

this is however theoretically valid under the assumption that the camera is positioned above the rear wheel axis of the vehicle. In the experimental section (6) will investigate under which conditions this approximation can still be adopted if the camera has an offset to the rear axis.

Now, we will see how the circular motion constraint reflects on the rotation and translation of the camera and on the parameterization of the essential matrix. In the following we will assume locally planar motion.

4.1 Parameterizing the Camera Motion

To understand the influence of the vehicle's nonholonomic constraints on the camera motion, we need to take into account two transformations: that between the camera and the vehicle and that between the two vehicle positions.

Let us assume that the camera is fixed somewhere on the vehicle[1] (with the origin in O_C, Fig. 2) with the axis $\mathbf{z_C}$ orthogonal to the plane of motion and $\mathbf{x_C}$ oriented perpendicularly to the back wheel axis. Observe that once the camera is installed on the vehicle the axes can be rearranged in the way above with a simple transformation of coordinates.

The origin O_V of the vehicle reference frame can be chosen arbitrarily. For convenience, we set O_V at the intersection of $\mathbf{x_C}$ with the back wheel axis, and $\mathbf{x_V}$ aligned with $\mathbf{x_C}$ (Fig. 2). We observed that by this choice the equations are notably simplified.

Following these considerations, the transformation $A_V^C = (\mathbf{R_V^C}, \mathbf{T_V^C})$ from the camera to the vehicle reference system can be written as $\mathbf{R_V^C} = \mathbf{I_{3\times 3}}$ and $\mathbf{T_V^C} = [-L, 0, 0]^T$, where L is the distance between the camera and the back wheel axis (Fig. 2).

If the vehicle undergoes perfect circular motion with rotation angle θ, then the direction of translation ϕ of the vehicle must satisfy the "circular motion constraint" $\phi = \theta/2$, which can be easily verified by trigonometry. Accordingly, the transformation between the first and second vehicle position $A_{V'}^V = (\mathbf{R_{V'}^V}, \mathbf{T_{V'}^V})$ can be written as:

$$\mathbf{R_{V'}^V} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad \mathbf{T_{V'}^V} = \rho \cdot \begin{bmatrix} \cos(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) \\ 0 \end{bmatrix} \tag{2}$$

where ρ is the vehicle displacement (Fig. 2). Following these considerations, the overall transformation $A_{C'}^C = (\mathbf{R_{C'}^C}, \mathbf{T_{C'}^C})$ between the first and second camera positions can be computed as a composition of the following three transformations, that is:

$$A_{C'}^C = A_V^C \circ A_{V'}^V \circ A_{C'}^{V'} = A_V^C \circ A_{V'}^V \circ A_V^{C^{-1}} \tag{3}$$

where we used $A_{C'}^{V'} = A_V^{C^{-1}}$. And from this, we obtain:

$$\mathbf{R_{C'}^C} = \mathbf{R_{V'}^V}, \quad \text{and} \quad \mathbf{T_{C'}^C} = \begin{bmatrix} L\cos(\theta) + \rho\cos(\frac{\theta}{2}) - L \\ \rho\sin(\frac{\theta}{2}) - L\sin(\theta) \\ 0 \end{bmatrix}. \tag{4}$$

4.2 Computing the Essential Matrix

Before going on, we would like to recall some knowledge about computer vision. Let $\mathbf{p} = (x, y, z)$ and $\mathbf{p}' = (x', y', z')$ be the image coordinates of a scene point seen from the two camera positions. Note, to make our approach independent of the camera model we use spherical image coordinates; therefore $\mathbf{p}$ and $\mathbf{p}'$ are the image points back projected onto a unit sphere (i.e. $\|\mathbf{p}\| = \|\mathbf{p}'\| = 1$). This is always possible when the camera is calibrated.

As known in computer vision (Hartley and Zisserman 2004), the two unknown camera positions and the image coordinates must verify the epipolar constraint

$$\mathbf{p}'^T \mathbf{E}\mathbf{p} = 0, \tag{5}$$

where $\mathbf{E}$ (called *essential matrix*) is defined as $\mathbf{E} = [\mathbf{T}]_\times \mathbf{R}$, where $[\mathbf{T}]_\times$ denotes the skew symmetric matrix

$$[\mathbf{T}]_\times = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix} \tag{6}$$

and $\mathbf{R}$ and $\mathbf{T} = [T_x, T_y, T_z]$ describe the relative pose between the camera positions (for our case $\mathbf{R} = \mathbf{R_{C'}^C}$ and $\mathbf{T} = \mathbf{T_{C'}^C}$).

The epipolar constraint (5) is very important because it allows us to estimate the relative camera pose from a set of

[1] Note that the camera does not necessarily have to be on the axis of symmetry of the vehicle.

image correspondences. Indeed, given the image points $\mathbf{p}$ and $\mathbf{p}'$ we can compute $\mathbf{E}$ from (5) and finally decompose $\mathbf{E}$ into $\mathbf{R}$ and $\mathbf{T}$ (Hartley and Zisserman 2004).

This said, we can now compute the essential matrix for our case using $\mathbf{E} = [\mathbf{T}^{\mathbf{C}}_{\mathbf{C}'}]_\times \mathbf{R}^{\mathbf{C}}_{\mathbf{C}'}$, that is:

$$\mathbf{E} = \begin{bmatrix} 0 & 0 & \rho\sin(\frac{\theta}{2}) - L\sin(\theta) \\ 0 & 0 & L + \rho\cos(\frac{\theta}{2}) - L\cos(\theta) \\ L\sin(\theta) + \rho\sin(\frac{\theta}{2}) & L - \rho\cos(\frac{\theta}{2}) - L\cos(\theta) & 0 \end{bmatrix}. \tag{7}$$

At this point, note that the essential matrix is notably simplified if $L = 0$, that is, when the camera is above the back wheel axis. Indeed, by substituting $L = 0$ into (7) we obtain:

$$\mathbf{E} = \rho \cdot \begin{bmatrix} 0 & 0 & \sin(\frac{\theta}{2}) \\ 0 & 0 & \cos(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & -\cos(\frac{\theta}{2}) & 0 \end{bmatrix}. \tag{8}$$

Finally, by imposing the epipolar constraint (5), we obtain the following homogeneous equation that needs to be satisfied by every pair of point correspondences $\mathbf{p}$, $\mathbf{p}'$:

$$\sin\left(\frac{\theta}{2}\right) \cdot (x'z + z'x) + \cos\left(\frac{\theta}{2}\right) \cdot (y'z - z'y) = 0. \tag{9}$$

Note, this equation depends only on the single parameter θ, showing that the relative camera motion can be recovered using a single feature correspondence.

4.3 Recovering θ

Given one point correspondence, the rotation angle θ can then be obtained from (9) as:

$$\theta = -2\tan^{-1}\left(\frac{y'z - z'y}{x'z + z'x}\right). \tag{10}$$

Conversely, given m image points, θ can be computed indirectly by solving linearly for the vector $[\sin(\frac{\theta}{2}), \cos(\frac{\theta}{2})]$ using Singular Value Decomposition (SVD). To this end, we first form a $m \times 2$ data matrix D, where each row is formed by the two coefficients of (9), that is:

$$\left[(x'z + z'x), (y'z - z'y)\right]. \tag{11}$$

The matrix D is then decomposed using SVD:

$$D_{m\times 2} = U_{m\times 2}\Lambda_{2\times 2}V_{2\times 2} \tag{12}$$

where the columns of $V_{2\times 2}$ contain the eigenvectors e_i of $D^T D$. The eigenvector $e^* = [\sin(\frac{\theta}{2}), \cos(\frac{\theta}{2})]$ corresponding to the minimum eigenvalue minimizes the sum of squares of the residuals, subject to $\|e^*\| = 1$. Finally, θ can be computed from e^*.

4.4 Discussion on Our Motion Model

To recap, we have shown that by fixing the camera in the optimal position $L = 0$ and under circular motion constraint the relative camera motion can be parameterized through a single feature correspondence.

In the next section we will see how this can be used for efficiently removing the outliers of the feature matching process. Then, we will investigate until which limit we can actually push L for our restrictive model to be still usable. Indeed, as observed in the expression of the essential matrix (7), when $L \neq 0$ the minimal model parameterization is 2 (θ and ρ), that is, at least two point correspondences are required to estimate the camera motion.[2] However, as we will point out in Sect. 6, our 1-point parameterization continues to be still a very good approximation in those cases where θ is small ($\theta < 10$ deg).

Finally, observe that the planar assumption and the circular motion constraint hold only locally, but because of the smooth motion of cars we found that this assumption actually holds still quite well also in the real situations; the performance will be shown in Sect. 6.

5 Outlier Removal: Two Approaches

Outlier removal is the most delicate process in camera pose estimation. The presence of a few outliers in the data may affect negatively the accuracy of the final motion estimate. Here, we describe two approaches for removing the outliers, which take advantage of our 1-point parameterization. Once the outliers are identified, the unconstrained motion estimate (6DoF) can be computed from all the remaining inliers using standard methods (Stewenius et al. 2006; Hartley and Zisserman 2004).

The two approaches explained here are based on RANSAC and histogram voting.

[2] Note that because ρ does not appear as a multiplicative factor in (7), this means that we can actually determine the absolute scale analytically from just two-point correspondences. This result was presented in our previous work (Scaramuzza et al. 2009).

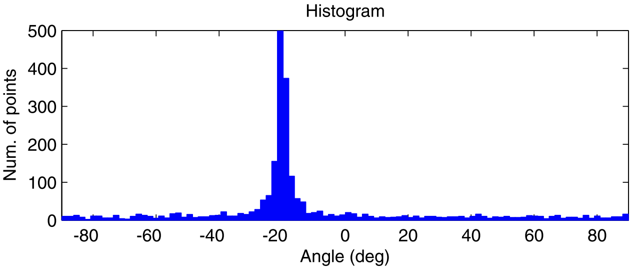


Fig. 3 A sample histogram from feature correspondences

5.1 1-Point RANSAC

The first step of our 1-point RANSAC consists in computing the relative motion out of one randomly chosen correspondence. To do this, we first use (10). The motion hypothesis is then generated using (2) (note, ρ can be arbitrarily set to 1). The second step is counting the inlier rate in each iteration, that is, the number of correspondences which satisfy the hypothesis. This can be done using the reprojection error. We used an error threshold of 1 pixel. Note, for an efficient computation of the reprojection error, some approximation exist, e.g. the *Sampson* distance (Hartley and Zisserman 2004) or the *directional error* by Oliensis (2002).

5.2 Histogram Voting

The possibility of estimating the motion using only one feature correspondence allows us to implement another algorithm for outlier removal which is much more efficient than the 1-point RANSAC as it requires no iterations. The algorithm is based on histogram voting: first, θ is computed from each feature correspondence using (10); then, a histogram H is built where each bin contains the number of features which count for the same θ. A sample histogram built from real data is shown in Fig. 3. When the circular motion model is well satisfied, the histogram has a very narrow peak centered on the best motion estimate θ^*, that is, $\theta^* = \text{argmax}\{H\}$.

In a first stage, we thought of selecting the inliers by taking all the features with θ within a given distance t from θ^*. We found that most of these points were indeed inliers, but there were still many missing points. Furthermore, the choice of t was not trivial. Therefore, the implemented solution consists again in using reprojection error, that is, we generate our motion hypothesis by substituting θ^* into (2) and use the reprojection error to identify all the inliers.

We also implemented a similar approach where, instead of computing θ^* as the argmax of the histogram, we set θ^* equal to the median of the distribution, that is, $\theta^* = \text{median}\{\theta_i\}$. The inliers are then found by using again the reprojection error. We found this method giving as good results as the argmax method and therefore we used this in our final implementation.

Compared with the 5-point RANSAC, the 1-point RANSAC and histogram voting method are the most efficient algorithms for removing the outliers. In all the tests, the computational time required to detect the inliers using the histogram voting method was in average 0.2 milliseconds, with a dataset of about 1600 points. The 1-point RANSAC found a successful solution in less than 7 iterations, requiring at most 1 millisecond. These tests were done with an Intel 2 GHz Dual Core laptop.

6 Experiments

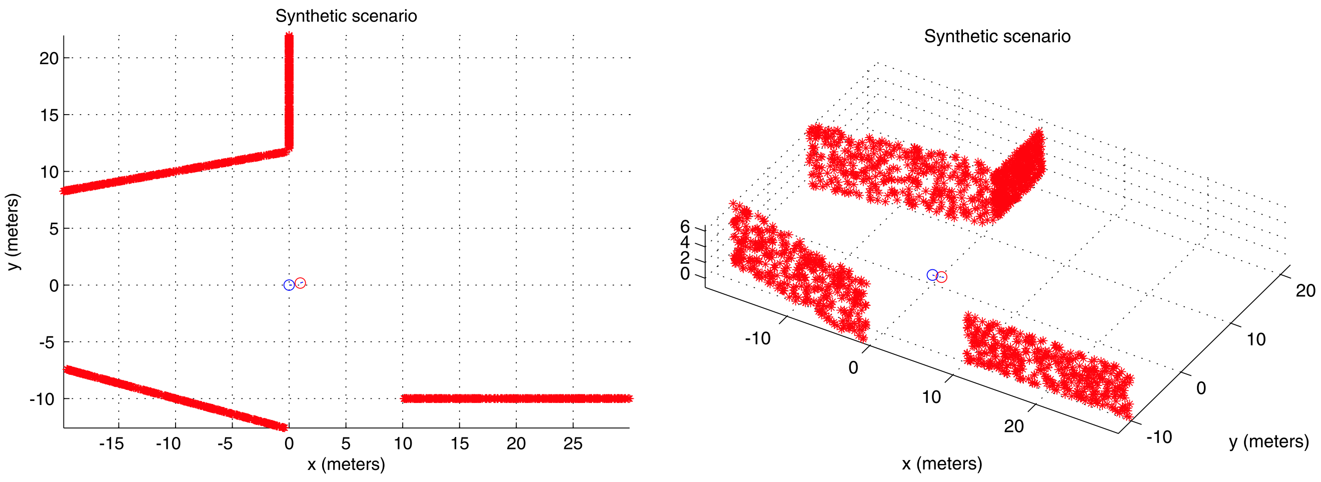
In this section, we will validate our motion model. The 1-point method and the histogram voting method will be compared with the 5-point algorithm by Nister (2003) and Stewenius et al. (2006), which is considered the standard in visual odometry (Lhuillier 2005; Nister et al. 2006; Tardif et al. 2008). In particular, we will investigate within which constraints our motion model is able to find as many (or more) correspondences as the 5-point method and when it becomes too restrictive.

As discussed in Sect. 4.4, in order to use our 1-point parameterization the camera needs to be installed above the back wheel axis, satisfying so the requirement $L = 0$. In this section, we will evaluate also under which motion conditions we can arbitrary fix the camera on the vehicle. The position of the camera is in fact of utmost importance in commercial automotive applications, where the camera is usually under the vehicle windscreen.

We will also evaluate the performance when the planarity constraint is not perfectly satisfied. For the 5-point method, we will use the implementation of the algorithm available at the authors' website. We will first compare the three algorithms on synthetic data and finally on real data.

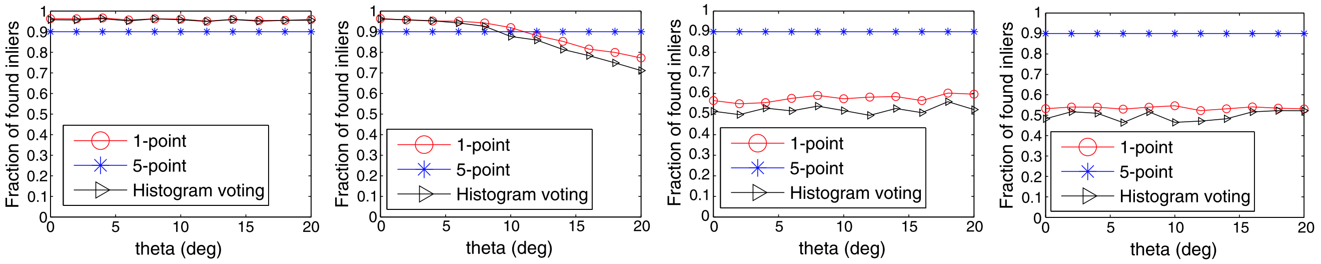
6.1 Generation of Synthetic Data

We investigate the performance of the algorithms in geometrically realistic conditions. In particular, we simulate a vehicle moving in urban canyons. Our scenario is depicted in Fig. 4. We set the first camera at the origin and randomise scene points uniformly inside several different planes, which stand for the facades of urban buildings. We used overall 1600 scene points, namely 400 on each plane. The second camera is positioned according to the motion direction of the vehicle which moves along circular trajectories about the instantaneous center of rotation. Therefore, the position of the second camera depends on the rotation angle θ, on the vehicle displacement ρ, and on the distance L of the camera from the center of the back wheels. These parameters are the same introduced in the previous sections.



(a) (b)

Fig. 4 Our synthetic scenario: (**a**) Top view, (**b**) 3D view



(a) Planar, $L/\rho = 0$ (b) Planar, $L/\rho = 1$ (c)Non-planar, $L/\rho = 0$ (d) Non-planar, $L/\rho = 1$

Fig. 5 Comparison between 1-point RANSAC, 5-point RANSAC, and histogram voting. Fraction of inliers versus θ

To make our analysis more realistic, we assume that the car can drive at a maximum speed of 50 Km/h and that the camera frame rate is 15 Hz (actually the one of our real camera). Accordingly, the maximum vehicle displacement between two frames is about 1 m. Therefore, as a default condition we set $\rho = 1$ m in all tests. The minimal distance of the scene to the camera was set at 10 m.

We also simulate feature location errors by introducing a noise parameter into the image data. We include a Gaussian perturbation in each image point with a standard deviation of 0.5 pixel in a 640×480 pixel image.

6.2 Comparison with 5-Point RANSAC

In this section, we evaluate the performance of our 1-point RANSAC and histogram voting with the standard 5-point RANSAC (Nister 2003; Stewenius et al. 2006). The performance is done by comparing the percentage of inliers found by the three methods, that is, the ratio between the found matches and the true number of inliers.

We evaluated the performance with respect to the rotation angle θ and the normalized camera offset L/ρ.[3] Since this would require to do the test for all the possible combinations of θ and L/ρ, we chose to show here only two extreme cases, that is, the optimal case $L/\rho = 0$ and the case $L/\rho = 1$. In fact, these two cases are those we tested also on our platform and therefore we decided to replicate them in simulation.

The average results, over one thousand trials, are shown in Fig. 5 for planar and non-perfectly planar motion respectively. For simulating a non-planar motion, we introduced a 0.1 m high step and a tilt angle of 1 deg. Note, we limited the range of θ in the simulations between 0 and 20 deg as this is what we experienced with the real data from our platform (see Fig. 6). Note, each plot in Fig. 5 corresponds to a different combination of motion (planar/non-planar) and camera settings ($L/\rho = 0$ and $L/\rho = 1$). For each combination, we generated one thousand trials; each trial consists in perturbing the image points with 0.5 pixel variance Gaussian

[3] Notice that in order to make our evaluation independent of the displacement of the vehicle, it is better to use an adimensional parameter.

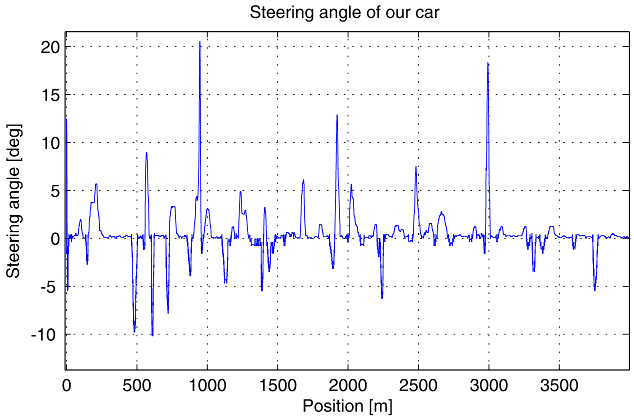


Fig. 6 Steering angle θ (deg) vs. traveled distance (m) read from our car. It is the angle the vehicle rotated between two consecutive frames

noise. Every dot in the plot shows the average over these one thousand trials for a given theta angle.

As observed in Fig. 5(a), for planar motion and $L/\rho = 0$, the performance of the algorithms stays constant with θ as expected. However, when $L/\rho = 1$, Fig. 5(b), the fraction of inliers found by the 1-point and histogram-voting methods decreases with θ, starting around $\theta = 10$ deg. When $\theta = 20$ deg, the two algorithms find 75% of the true inliers. The performance of the 5-point method stays conversely constant with θ regardless of L/ρ. The 5-point method indeed does not assume motion constraints.

For non-perfectly planar motion, Figs. 5(c)–(d), the performance of the 1-point and histogram-voting methods decreases notably, with only 50% of the inliers detected.

6.3 Number of RANSAC Iterations

We repeated the experiments presented in the previous section by varying also the percentage of outliers in the datapoints from 10% up to 90%. The results were the same as introduced in Fig. 5 regardless of the number of outliers in the datapoints. However, the number of RANSAC iterations needed to find the largest set of inliers increased exponentially with the percentage of outliers.[4] For instance, when the outliers were 70% of the datapoints, the 5-point RANSAC needed more than 1500 iterations. A comparison of the number of iterations needed to find the largest set of inliers as a function of the percentage of outliers is shown in Fig. 7. These results are the average over different trials. Note, here we also added a comparison with the 2-point RANSAC.

As predicted by (1), the number of iterations of the 1-point and 5-point RANSAC increases exponentially with

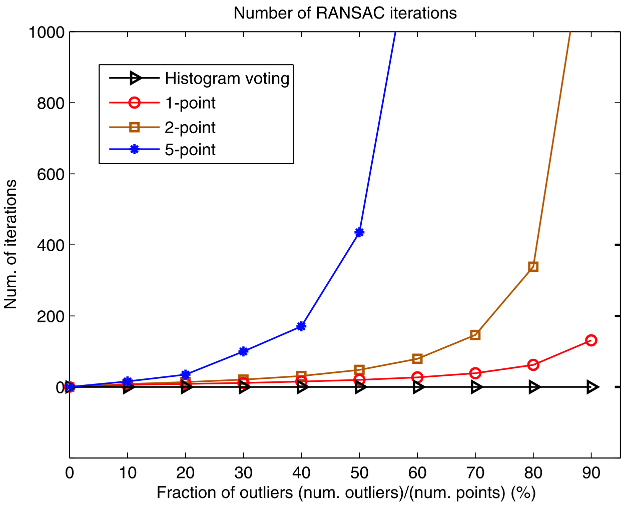


Fig. 7 Number of RANSAC iterations versus fraction of outliers

the fraction of outliers. But the number of iterations of the 1-point is greatly smaller than that of the 5-point. For instance, in the worse case, with 90% of outliers, the 5-point needed more than 2000 iterations while the 1-point method required only 90 iterations. The histogram-voting method does not require iterations but is shown here just for comparison.

6.4 Experiments on Real Data

Note, the equations and results derived in this paper are valid for both perspective and omnidirectional cameras. To show the generality of the approach we decided to use an omnidirectional camera.

(1) *Data Acquisition*: The method described in this paper has been successfully tested on a real vehicle (Fig. 9). Our omnidirectional camera is composed of a hyperbolic mirror (KAIDAN 360 One VR) and a digital color camera (SONY XCD-SX910, image size 640 × 480 pixels).

For the purpose of this paper, we tested the algorithms with the camera in two different positions: camera above the back wheel axis ($L = 0$) and camera above the front wind screen as in Fig. 9 ($L = 1$ m). To do this, we collected two datasets with the camera at different positions. We used the maximum frame rate of this camera, which is 15 Hz but sometimes we noticed that the frame rate decreased below 10 Hz because of the memory sharing on the on-board computers. For calibrating the camera we used the toolbox described in Scaramuzza et al. (2006) and available from the authors' website. The vehicle speed ranged between 0 and 45 Km/h.

The dataset was taken in normal traffic in the city center of Zurich during a 3 Km trajectory (Fig. 13). Therefore, many pedestrians, moving trams, buses, and cars were also present. Point correspondences were extracted using the Harris detector (Harris and Stephens 1988).

[4] As a stopping criterion, here we used the method proposed in Hartley and Zisserman (2004), which adaptively estimates the fraction of outliers in the data and computes accordingly the number of iterations required using (1).

6.4.1 Inlier ratio

To evaluate the performance on real data, we compare the percentage of inliers found by the three methods under different conditions which are: $L = 0$, $L = 1$ m, flat road, non-perfectly flat road, straight and curving path, low frame rate. Because we cannot show the results for the all 4000 images in our dataset, we decided to show them only for some selected paths. The results of the comparison are presented in Fig. 8 while the paths they refer to are shown in Fig. 13. As observed in Fig. 8, the performance of the 1-point and histogram-voting methods compare very well with the 5-point method for the first four cases (a–b–c–d). The performance of the two algorithms is slightly lower in the fifth path (Fig. 8(e)) where the camera frame rate drops to 2.5 Hz. We can justify this by observing that our restrictive motion model holds only locally and it is therefore important that the displacement of the vehicle between two consecutive frame be small. The performance drastically decreases at some point in the sixth path where the car is going downhill on a slightly twisting road.

By inspecting the performance for the all dataset, we found that the percentage of inliers of the 1-point and histogram-voting methods differed from that of the 5-point by less than 10% in 80% of the cases. This is clearly quantified in Fig. 10, which shows the histogram of the relative difference (%) between the inlier count of the 1-point and the 5-point algorithm over all images. When the difference was larger than 10%, we found that this was due to sudden jumps of the frame-rate or to non-perfect planarity of the road. To verify this last statement quantitatively, we measured the planarity of the motion estimated by the 5-point algorithm. The planarity of the motion was characterized both in terms of the estimated tilt angle Ω and in terms of the estimated camera displacement Z along z. For every pair of consecutive images, we computed both Ω and Z and measured the ratio $\frac{\#inliers_{1p}}{\#inliers_{5p}}$. The relation between the non-planarity of the estimated motion and the inlier ratio is shown in Figs. 11 and 12. These plots depict mean and standard deviation of the inlier ratio computed within predefined intervals of Ω and Z, respectively. As observed, a reduced number of inliers in the 1-point algorithm occurs when the planar motion assumption is violated. Furthermore, the less planar the motion, the smaller the number of inliers. This result is perfectly in line with what we predicted in simulation in Sect. 6.2.

Despite this, from Fig. 10 we can see that our restrictive motion model is a good approximation of the motion of the car. Furthermore, in the all experiment we found that the 1-point and the histogram-voting method performed the same. However, we also observed that in presence of low frame rate or non-planar motion the performance of the histogram-voting was slightly lower. Regarding the computational cost, during all the experiment we found that the 1-point RANSAC required at most 7 iterations while the 5-point RANSAC needed from 500 up to 2000 iterations.

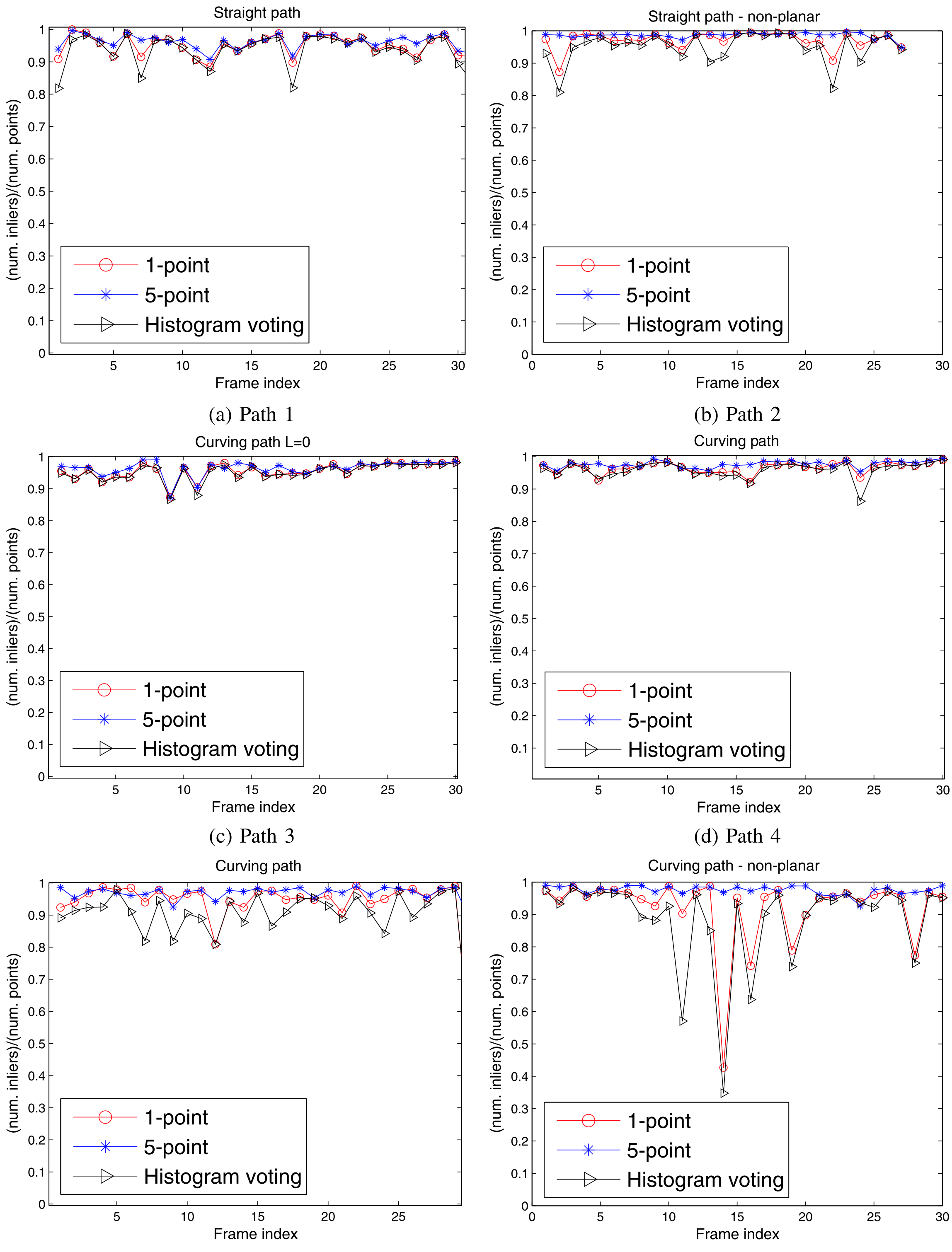
(3) *Visual odometry*: To evaluate the quality of point correspondences output by our proposed methods, we implemented a motion estimation algorithm and we run it on the entire 3 Km dataset. For this experiment, we implemented a very simple, incremental motion estimation algorithm, which means, we only computed the motion between consecutive frames (e.g. two-view structure-from-motion). Note, we did not use the previous poses and structure to refine the current estimate. Furthermore, we did not use bundle-adjustment. For removing the outliers, we used one of our proposed methods. From the remaining inliers, the relative pose was then estimated using the motion-estimation algorithm in Stewenius et al. (2006), which provides unconstrained 6DoF motion estimates. The absolute scale between consecutive poses was measured by simply reading the speed of the car from the vehicle CAN-bus and multiplying it by the time interval between the two frames. The recovered trajectory using the histogram-voting method for outlier-removal is shown in Fig. 13 overlaid on a satellite image. Note that this algorithm run at 400 fps.

Figure 14 shows instead the comparison among the visual odometry paths computed with histogram-voting, 1-point, and 5-point RANSAC. As the reader can see, the trajectory estimated by the histogram voting method differs very little from that estimated with the 1-point RANSAC. Furthermore, both methods seem to outperform the 5-point RANSAC. This result should not surprise the reader. Indeed, let us remind that we did not use bundle adjustment, which obviously would largely reduce the accumulated drift. However, it is also important to point out that sometimes the found inliers are not the largest RANSAC consensus, meaning that more iterations would have actually been necessary. Additionally, this result points out that even though for most of the frames the 5-point RANSAC finds a little more inliers than the 1-point RANSAC, the 1-point RANSAC and the histogram voting methods output "better" inliers, in that they favour the underlying motion model.

7 Conclusion

In this paper, we have shown that by exploiting the nonholonomic constraints of a wheeled vehicle it is possible to parameterize the motion with a single feature correspondence. This parameterization is the smallest possible and resulted in the two most efficient algorithms for removing outliers.

We have seen that for car-like and differential drive robots this 1-point parameterization is satisfied only by fixing the camera above the back wheel axis ($L = 0$). However, in the experimental section we have demonstrated that



(a) Path 1

(b) Path 2

(c) Path 3

(d) Path 4

(e) Path 5

(f) Path 6

Fig. 8 Comparison 1-point, 5-point, and histogram voting. Percentage of good matched versus frame number. (**a**) Straight path, flat road, $L = 1$ m. (**b**) Straight path, non-perfectly flat (e.g. crossing the tram rail ways), $L = 1$ m. (**c**) Curving path, flat road, $L = 0$ m. (**d**) Curving path, flat road, $L = 1$ m. (**e**) Curving path, flat road, $L = 1$ m, camera frame rate 2.5 Hz. (**f**) Curving path, non-perfectly flat road (going down hill with slightly twisting road), $L = 1$ m

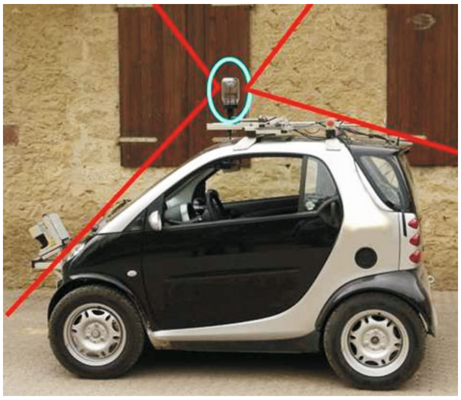


Fig. 9 Our vehicle equipped with the omnidirectional camera. The field of view is highlighted

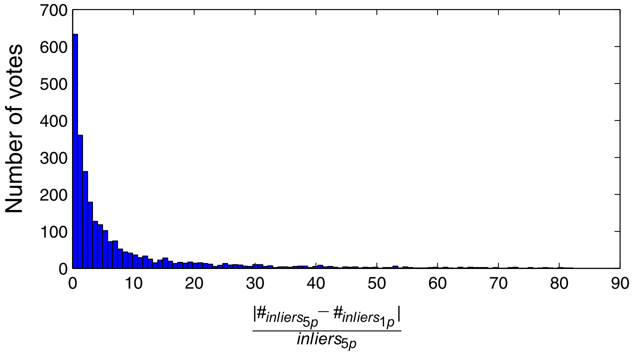


Fig. 10 Histogram of the relative difference (%) between the inlier count of the 1-p and the 5-p algorithm over all consecutive image pairs. This difference is computed as $\frac{|\#_{inliers_{5p}} - \#_{inliers_{1p}}|}{\#_{inliers_{5p}}}$. As observed, the percentage of inliers of the 1-point method differs from that of the 5-point by less than 10% in 80% of the cases. The histogram voting method gave the same performance and therefore it is not shown here

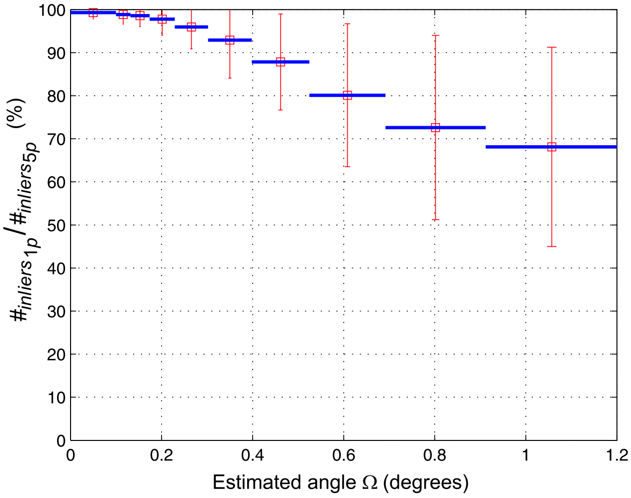


Fig. 11 Effect of the estimated tilt angle Ω on the ratio between the inlier count of the 1-point and the inlier count of the 5-point algorithm: ($\#_{inliers_{1p}} / \#_{inliers_{5p}}$). Mean and standard deviation of this ratio are computed within predefined intervals of Ω

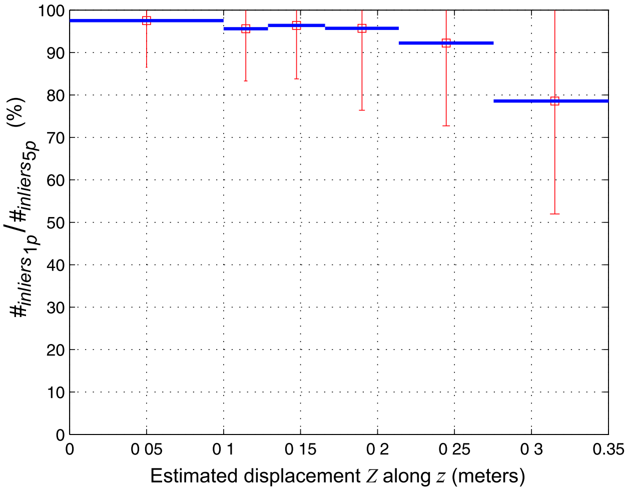


Fig. 12 Effect of the estimated displacement Z along z on the ratio between the inlier count of the 1-point and the inlier count of the 5-point algorithm: ($\#_{inliers_{1p}} / \#_{inliers_{5p}}$). Mean and standard deviation of this ratio are computed within predefined intervals of Z

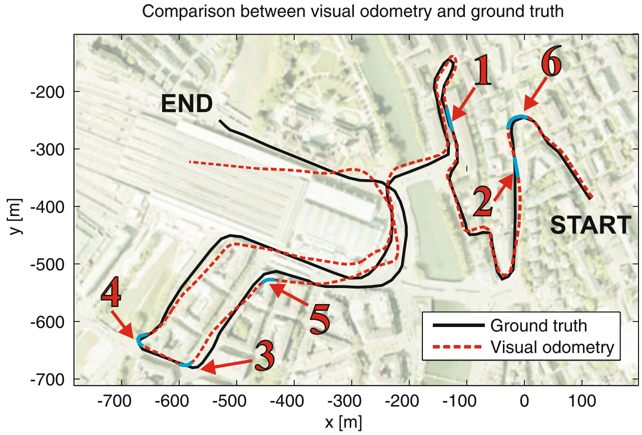


Fig. 13 (Color online) Comparison between visual odometry (*red dashed line*) and ground truth (*black solid line*). The entire trajectory is 3 Km long. The numbers correspond to the sequences analyzed in Fig. 8. Blue lines mark starting and ending points of each sequence

also for the case $L \neq 0$ our restrictive model is still suitable under the constraint that the rotation angle θ between two camera poses is small. In particular we have shown that in most cases our 1-point and histogram-voting methods perform similarly to the standard 5-point method, finding almost the same number of inliers. Finally, we showed the quality of the output correspondences by recovering visually the trajectory of the car.

Both the simulated and real experiments have pointed out that our restrictive model is a suitable approximation of the real motion of the vehicle provided that the road is nearly flat and the frame-rate is high (e.g. $>$ 10 Hz at 50 Km/h). This is because the circular motion model holds only locally. When the conditions for the validity of the model are not satis-

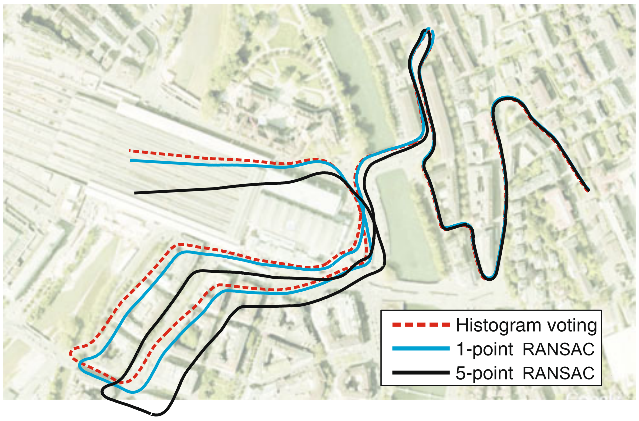


Fig. 14 (Color online) Comparison between visual odometry trajectories using the three different methods for outlier removal: histogram-voting (*red dashed line*), 1-point RANSAC (*cyan solid line*), and 5-point RANSAC (*black solid line*)

fied this reflects in a reduced number of inliers found by the 1-point and histogram voting methods. However, when this happens the problem can be easily overcome by switching to the standard 5-point RANSAC. Failure modes in the 1-point methods can be easily detected by looking at the histogram distribution. In fact, when the local circular planar motion is well verified, this reflects in a narrow histogram with a very distinguishable peak. Conversely, when our motion assumption does not hold, the resulting histogram appears wider. In these cases, looking at the variance of the distribution provides an easy way to switch between the 1-point and 5-point approaches.

References

Clemente, L. A., Davison, A. J., Reid, I., Neira, J., & Tardos, J. D. (2007). Mapping large loops with a single hand-held camera. In *Robotics science and systems*.

Corke, P. I., Strelow, D., & Singh, S. (2004). Omnidirectional visual odometry for a planetary rover. In *IROS*.

Davison, A. (2003). Real-time simultaneous localisation and mapping with a single camera. In *International conference on computer vision*.

Deans, M. C. (2002). *Bearing-only localization and mapping*. PhD thesis, Carnegie Mellon University.

Faugeras, O., & Maybank, S. (1990). Motion from point matches: multiplicity of solutions. *International Journal of Computer Vision*, *4*, 225–246.

Fischler, M. A., & Bolles, R. C. (1981). RANSAC random sampling concensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of ACM*, *26*, 381–395.

Goecke, R., Asthana, A., Pettersson, N., & Petersson, L. (2007). Visual vehicle egomotion estimation using the Fourier-Mellin transform. In *IEEE intelligent vehicles symposium*.

Harris, C., & Stephens, M. (1988). A combined corner and edge detector. In *Fourth alvey vision conference* (pp. 147–151).

Hartley, R., & Zisserman, A. (2004). *Multiple view geometry in computer vision* (2nd ed.). Cambridge: Cambridge University Press ISBN:0521540518.

Jung, I., & Lacroix, S. (2005). Simultaneous localization and mapping with stereovision. In *Robotics research: the 11th international symposium*.

Kruppa, E. (1913). Zur ermittlung eines objektes aus zwei perspektiven mit innerer orientierung. In *Abt. IIa.: Vol. 122. Sitz.-Ber. Akad. Wiss., Wien, Math. Naturw. Kl.* (pp. 1939–1948).

Lacroix, S., Mallet, A., Chatila, R., & Gallo, L. (1999). Rover self localization in planetary-like environments. In *International symposium on articial intelligence, robotics, and automation for space (i-SAIRAS)* (pp. 433–440).

Lemaire, T., & Lacroix, S. (2007). Slam with panoramic vision. *Journal of Field Robotics*, *24*, 91–111.

Lhuillier, M. (2005). Automatic structure and motion using a catadioptric camera. In *IEEE workshop on omnidirectional vision*.

Longuet-Higgins, H. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature*, *293*, 133–135.

Maimone, M., Cheng, Y., & Matthies, L. (2007). Two years of visual odometry on the mars exploration rovers: Field reports. *Journal of Field Robotics*, *24*, 169–186.

Milford, M. J., & Wyeth, G. (2008). Single camera vision-only slam on a suburban road network. In *IEEE international conference on robotics and automation, ICRA'08*.

Milford, M., Wyeth, G., & Prasser, D. (2004). Ratslam: A hippocampal model for simultaneous localization and mapping. In *International conference on robotics and automation, ICRA'04*.

Moravec, H. (1980). *Obstacle avoidance and navigation in the real world by a seeing robot rover*. PhD thesis, Stanford University.

Nister, D. (2003). An efficient solution to the five-point relative pose problem. In *CVPR03*.

Nister, D. (2005). Preemptive ransac for live structure and motion estimation. *Machine Vision and Applications*, *16*, 321–329.

Nister, D., Naroditsky, O., & Bergen, J. (2006). Visual odometry for ground vehicle applications. *Journal of Field Robotics*

Oliensis, J. (2002). Exact two-image structure from motion. *PAMI*.

Ortin, D., & Montiel, J. M. M. (2001). Indoor robot motion based on monocular images. *Robotica*, *19*, 331–342.

Philip, J. (1996). A non-iterative algorithm for determining all essential matrices corresponding to five point pairs. *Photogrammetric Record*, *15*, 589–599.

Pizarro, O., Eustice, R., & Singh, H. (2003). Relative pose estimation for instrumented, calibrated imaging platforms. In *DICTA*.

Scaramuzza, D., & Siegwart, R. (2008). Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. *IEEE Transactions on Robotics*, Special Issue on Visual SLAM, *24*.

Scaramuzza, D., Martinelli, A., & Siegwart, R. (2006). A toolbox for easy calibrating omnidirectional cameras. In *IEEE international conference on intelligent robots and systems (IROS 2006)*.

Scaramuzza, D., Fraundorfer, F., Pollefeys, M., & Siegwart, R. (2008). Closing the loop in appearance-guided structure-from-motion for omnidirectional cameras. In *Eighth workshop on omnidirectional vision (OMNIVIS'08)*.

Scaramuzza, D., Fraundorfer, F., Pollefeys, M., & Siegwart, R. (2009). Absolute scale in structure from motion from a single vehicle mounted camera by exploiting nonholonomic constraints. In *International conference on computer vision*.

Siegwart, R., Nourbakhsh, I., & Scaramuzza, D. (2011). *Introduction to autonomous mobile robots* (2nd ed.). Cambridge: MIT Press.

Stewenius, H., Engels, C., & Nister, D. (2006). Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, *60*, 284–294.

Tardif, J., Pavlidis, Y., & Daniilidis, K. (2008). Monocular visual odometry in urban environments using an omnidirectional camera. In *IEEE IROS'08*.