# Performance Evaluation of 1-Point-RANSAC Visual Odometry

**Davide Scaramuzza**
*GRASP Lab, University of Pennsylvania, Philadelphia, Pennsylvania 19104*
*e-mail: davide.scaramuzza@ieee.org*

Received 17 November 2010; accepted 5 July 2011

Monocular visual odometry is the process of computing the egomotion of a vehicle purely from images of a single camera. This process involves extracting salient points from consecutive image pairs, matching them, and computing the motion using standard algorithms. This paper analyzes one of the most important steps toward accurate motion computation, which is outlier removal. The random sample consensus (RANSAC) has been established as the standard method for model estimation in the presence of outliers. RANSAC is an iterative method, and the number of iterations necessary to find a correct solution is exponential in the minimum number of data points needed to estimate the model. It is therefore of utmost importance to find the minimal parameterization of the model to estimate. For unconstrained motion [six degrees of freedom (DoF)] of a calibrated camera, this would be five correspondences. In the case of planar motion, the motion model complexity is reduced (three DoF) and can be parameterized with two points. In this paper we show that when the camera is installed on a nonholonomic wheeled vehicle, the model complexity reduces to two DoF and therefore the motion can be parameterized with a single-point correspondence. Using a single-feature correspondence for motion estimation is the lowest model parameterization possible and results in the most efficient algorithm for removing outliers, which we call 1-point RANSAC. To support our method, we run many experiments on both synthetic and real data and compare the performance with state-of-the-art approaches and with different vehicles, both indoors and outdoors. © 2011 Wiley Periodicals, Inc.

## 1. INTRODUCTION

Visual odometry is the problem of estimating the egomotion of a vehicle from onboard-camera images. Several works have been recently produced using both stereo and monocular cameras (Goecke, Asthana, Pettersson, & Petersson, 2007; Maimone, Cheng, & Matthies, 2007; Milford & Wyeth, 2008; Nister, Naroditsky, & Bergen, 2006; Scaramuzza & Siegwart, 2008; Tardif, Pavlidis, & Daniilidis, 2008). Basically, visual odometry operates by incrementally computing the motion between consecutive frames. This is done by extracting salient points (such as Harris, FAST, SIFT) from both images and matching them according to some similarity measure. However, matched points are usually contaminated by outliers, that is, wrong data associations. Outliers must be carefully removed so that the motion can be estimated accurately (Figure 1).

The 5-point random sample consensus (RANSAC) (Nister, 2003) has been established as the standard method for motion estimation in the presence of outliers. The drawback of this algorithm is that the number of iterations necessary to find a correct solution grows exponentially with the number of outliers. In some cases, the 5-point RANSAC can require up to 1,000 iterations for data collected from a vehicle in an urban environment. Because of this, several works have been produced in an endeavor to reduce the number of iterations (Chum & Matas, 2005; Nister, 2005; Raguram, Frahm, & Pollefeys, 2009).

In this paper, we show that there exists a class of vehicles (e.g., cars, bicycles, differential-drive robots) for which the motion can be parameterized with a single-feature correspondence. This is made possible by exploiting the nonholonomic constraints of the vehicle. The main advantages and consequences of this parameterization are that motion can be estimated in the presence of very few features—where standard algorithms would fail—and that two very efficient methods for removing the outliers can be devised, which are 1-point RANSAC and histogram voting. Once the outliers are removed, the motion can be computed in six degrees of freedom (DoF) using all the inliers.

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

## 2. RELATED WORK ON VISUAL ODOMETRY

Most of the work in estimating vehicle motion using vision (also called visual odometry) has been produced using stereo cameras (Jung & Lacroix, 2005; Lacroix, Mallet,

View this article online at wileyonlinelibrary.com • DOI: 10.1002/rob.20411

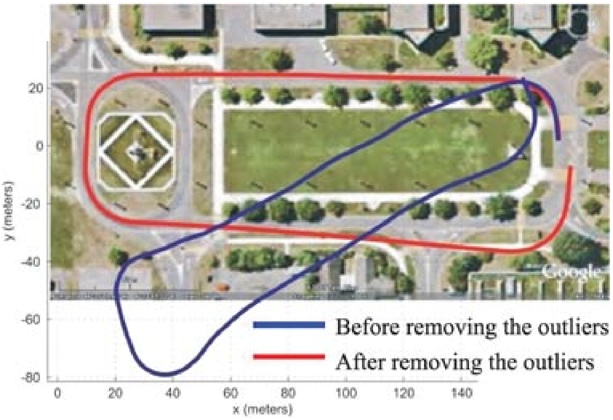


**Figure 1.** Comparison between visual odometry trajectories estimated before and after removing the outliers.

Chatila, & Gallo, 1999; Maimone et al., 2007; Moravec, 1980; Nister et al., 2006). Nevertheless, visual odometry methods for outdoor applications that use a single camera have also been produced. The problem of recovering relative camera poses and three-dimensional (3D) structure from a set of monocular images has been studied for many years and is known in the computer vision community as "structure from motion" (SFM) (Hartley & Zisserman, 2004). Successful results with a single camera and over long distances (from hundreds of meters up to kilometers) have been obtained in the past decade using both perspective and omnidirectional cameras (see Corke, Strelow, & Singh, 2005; Goecke et al., 2007; Lhuillier, 2005; Milford & Wyeth, 2008; Nister et al., 2006; Scaramuzza & Siegwart, 2008; Tardif et al., 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 these two.

In the first category are the works of Corke et al. (2005), Lhuillier (2005), Nister et al. (2006), Ortin and Montiel (2001), and Tardif et al. (2008). In Nister et al. (2006), the authors dealt with the case of a stereo camera, but they also provided a monocular solution. For removing the outliers, they used the 5-point RANSAC algorithm from one of the same authors (Nister, 2003). In Corke et al. (2005), the authors provided an approach for monocular visual odometry based on omnidirectional imagery from a catadioptric camera. Because the motion baseline was very small, outliers were removed if the pixel distance between the observed features and the predicted features was larger than a given threshold. However, this technique is not a good option when the motion baseline is large, as for general ground vehicles. In Lhuillier (2005), the author presented an approach-based bundle adjustment to recover both the motion and the 3D map. Again, they used the 5-point RANSAC in Nister (2003) for removing the outliers. In Tardif et al. (2008), the authors 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). In Ortin and Montiel (2001), the authors described an approach for planar-motion estimation that is based on a 2-point RANSAC. This 2-point parameterization was made possible by the fact that for planar motion correspondences of as few as two points are required to estimate the motion. Given the complexity of the equations, the authors determined the solution iteratively with the Newton–Raphson method.

Among the appearance-based or hybrid approaches are the works of Goecke et al. (2007), Milford and Wyeth (2008), and Scaramuzza and Siegwart (2008). In Goecke et al. (2007), the authors used the Fourier–Mellin transform for registering perspective images of the ground plane taken from a car. In Milford and Wyeth (2008), the authors 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, Wyeth, & Prasser, 2004). However, appearance-based approaches alone are not very robust to occlusions. For this reason, in our previous works (Scaramuzza, Fraundorfer, Pollefeys, & Siegwart, 2008; Scaramuzza & Siegwart, 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 SFM 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 recent years successful results have been obtained also using single cameras (see Civera, Grasa, Davison, & Montiel, 2010; Clemente, Davison, Reid, Neira, & Tardos, 2007; Davison, 2003; Deans, 2002; Eade & Drummond, 2007; Handa, Chli, Strasdat, & Davison, 2010; Klein & Murray, 2008; Lemaire & Lacroix, 2007). In particular, in Handa et al. (2010) the authors propose an active matching technique based on a probabilistic framework. In Civera et al. (2010) the authors propose a combination of 1-point RANSAC within an extended Kalman filter (EKF) that uses the available prior probabilistic information from the EKF in the RANSAC model hypothesis stage. However, our work is completely different as we do not use any prior probabilistic information. Our approach relies only on the use of the vehicle nonholonomic constraints. The basic idea behind the current paper was already presented in our previous work in Scaramuzza, Fraundorfer, and Siegwart (2009) and in Scaramuzza (2011). Our previous work was, however, based on the assumption that the camera is positioned on the rear-wheel axis. In this paper, we provide a complete and deep evaluation of the influence of the camera position on the estimate of the relative motion and we compare our approach with standard algorithms such as the 2-point and the 5-point RANSAC. Furthermore, we test the algorithm on additional data sets and vehicles, in both indoor and outdoor environments.

## 3. MINIMAL MOTION MODELS

For unconstrained motion (six DoF) of a calibrated camera, the minimum number of point correspondences required for solving the relative pose problem is five [see the 5-point algorithm in Nister (2003) and Stewenius, Engels, and Nister (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 in Kruppa (1913) to have at most 11 solutions. This was later improved by Faugeras and Maybank (1990), showing that there are at most 10 solutions, but the method found only in 2006 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 to very efficient and easy-to-implement algorithms. The 8-point algorithm (Longuet-Higgins, 1981) has a linear solver for a unique solution, whereas the 7-point method (Hartley and Zisserman, 2004) leads to up to three solutions. The 6-point method (Philip, 1996; Pizarro, Eustice, & Singh, 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.

In every situation in which a model has to be estimated from given data, we have to deal with outliers. The RANSAC (Fischler & Bolles, 1981) has been established as the standard method for model estimation in the presence of outliers. SFM is one application of the RANSAC scheme. The estimated model is the motion (**R**,**T**) and 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 to verify these hypotheses on the other data points. The hypothesis that shows the highest consensus with the other data is selected as the solution. The number of subsets (iterations) $N$ that is necessary to guarantee that a correct solution is found can be computed by

$$N = \frac{\log(1-p)}{\log[1-(1-\epsilon)^s]}, \tag{1}$$

where $s$ is the number of minimal data points, $\epsilon$ is the percentage of outliers in the data points, and $p$ is the requested probability of success (Fischler & 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 (six DoF) of a calibrated camera, this would be five correspondences. Using the 6-, 7-, or 8-point method would increase the number of necessary iterations and therefore slow 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 (three DoF) and can be parameterized with two points as described in Ortin and Montiel (2001).

**Table I.** Number of RANSAC iterations.

| Minimal number of data points ($s$) | Number of iterations ($N$) |
|---|---|
| 8 | 1,177 |
| 7 | 587 |
| 6 | 292 |
| 5 | 145 |
| 2 | 16 |
| 1 | 7 |

For wheeled vehicles we will show in Section 4 that an even more restrictive motion model can be chosen that allows us to parameterize the motion with only one 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 that requires no iteration can be devised.

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

## 4. PARAMETERIZING THE MOTION WITH ONE-POINT CORRESPONDENCE

For a wheeled vehicle to exhibit rolling motion, a point must exist around which each wheel of the vehicle follows a circular course (Siegwart, Nourbakhsh, & Scaramuzza,

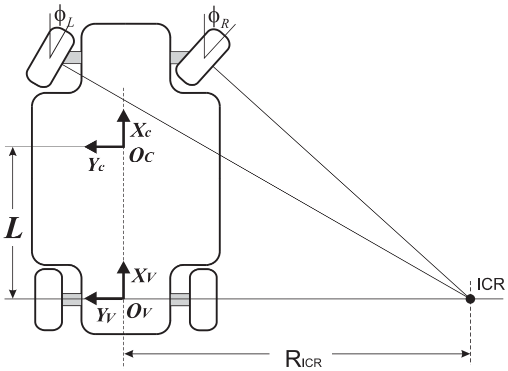


**Figure 2.** General Ackermann steering principle.

2004). This point is known as the instantaneous center of rotation (ICR) and can be computed by intersecting all the roll axes of the wheels (Figure 2). This property holds for any robot, in particular for car-like and differential-drive robots. For cars the existence of the ICR is ensured by the Ackermann steering principle (Siegwart et al., 2004). This principle ensures a smooth movement of the vehicle by applying different steering angles to the inner and outer front wheel while turning (see Figure 2).

As the reader can perceive, the motion of a camera fixed on the vehicle can then be locally described with circular motion (note that rectilinear motion can be represented along a circle with an infinite radius of curvature). This constraint reduces the DoF 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, this is, however, theoretically valid under the assumption that the camera is positioned above the rear-wheel axis of the vehicle. In Section 6 we will evaluate 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 (with the origin in $O_C$; Figure 3) with the axis $\mathbf{z_C}$ orthogonal to the plane of motion and $\mathbf{x_C}$ oriented perpendicularly to the rear-wheel axis. Observe that once the camera is installed on the vehicle, the axes can be rearranged in the above way with a simple transformation of coordinates.

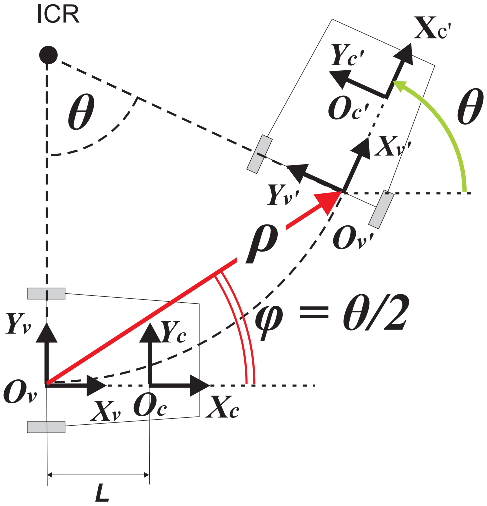


**Figure 3.** Relation between camera axes in circular motion.

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 rear-wheel axis and $\mathbf{x_V}$ aligned with $\mathbf{x_C}$ (Figure 3). 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 $\mathbf{R_V^C} = \mathbf{I}_{3\times3}$ and $\mathbf{T_V^C} = [-L, 0, 0]^T$, where $L$ is the distance between the camera and the back-wheel axis (Figure 3).

If the vehicle undergoes perfect circular motion with rotation angle $\theta$, then the direction of translation $\phi$ of the vehicle must satisfy the "circular motion constraint" $\phi = \theta/2$, which can be easily verified by goniometrics. 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

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

where $\rho$ is the vehicle displacement (Figure 3). 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 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 $A^{V'}_{C'} = A^{C^{-1}}_{V}$. And from this, we obtain

$$\mathbf{R}^{\mathbf{C}}_{\mathbf{C}'} = \mathbf{R}^{\mathbf{V}}_{\mathbf{V}'}, \text{ and } \mathbf{T}^{\mathbf{C}}_{\mathbf{C}'} = \begin{bmatrix} L\cos(\theta) + \rho\cos\left(\frac{\theta}{2}\right) - L \\ \rho\sin\left(\frac{\theta}{2}\right) - L\sin(\theta) \\ 0 \end{bmatrix}. \quad (4)$$

### 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 that to make our approach independent of the camera model, we use spherical image coordinates; therefore $\mathbf{p}$ and $\mathbf{p}'$ are the image points backprojected onto a unit sphere (i.e., $\|\mathbf{p}\| = \|\mathbf{p}'\| = 1$). This is always possible when the camera is calibrated.

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

$$\mathbf{p}'^{T}\mathbf{E}\mathbf{p} = 0, \quad (5)$$

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

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

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

The epipolar constraint (5) is very important because it allows us to estimate the relative camera pose from a set of image correspondences. Indeed, given the image points $\mathbf{p}$ and $\mathbf{p}'$ we can compute $\mathbf{E}$ from Eq. (5) and finally decompose $\mathbf{E}$ into $\mathbf{R}$ and $\mathbf{T}$ (Hartley & Zisserman, 2004).

That 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,

$$\mathbf{E} = \begin{bmatrix} 0 & 0 & \sin\left(\frac{\theta}{2}\right) - \frac{L}{\rho}\sin(\theta) \\ 0 & 0 & \cos\left(\frac{\theta}{2}\right) + \frac{L}{\rho}[1 - \cos(\theta)] \\ \frac{L}{\rho}\sin(\theta) + \sin\left(\frac{\theta}{2}\right) & \frac{L}{\rho}[1 - \cos(\theta)] - \cos\left(\frac{\theta}{2}\right) & 0 \end{bmatrix}. \quad (7)$$

As can be observed, $\mathbf{E}$ depends on two parameters, namely $\theta$ and the ratio $L/\rho$.[1] Because $L/\rho$ is unknown, $\mathbf{E}$ needs a minimum of two point correspondences to be computed, and therefore this problem becomes equivalent to that of estimating the essential matrix under general planar motion (Hartley & Zisserman, 2004). However, we observe that $\mathbf{E}$ gets notably simplified if we pose $L = 0$, that is, when the camera is above the rear-wheel axis.[2] Indeed, by substituting $L = 0$ into (7) we obtain

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

We can observe that $\mathbf{E}$ now depends only on $\theta$, since $\rho$ appears to be just a multiplicative factor. This implies that we need just one point correspondence to estimate $\mathbf{E}$. Indeed, if we now impose the epipolar constraint (5), we obtain the following homogeneous equation that needs to be satisfied by every pair of point correspondences $\mathbf{p}$, $\mathbf{p}'$:

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

Again, we can see that this equation depends only on the single parameter $\theta$, showing that the relative camera motion can be recovered using a single-feature correspondence.

From now on we will assume that the camera is positioned above the rear-wheel axis and therefore $L = 0$. In the experimental section (Section 6) we will investigate under which conditions this approximation can still be adopted if the camera has an offset to the rear axis (i.e., $L \neq 0$).

### 4.3. Recovering $\theta$

Given one-point correspondence, the rotation angle $\theta$ can then be obtained from Eq. (9) as

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

Conversely, given $m$ image points, $\theta$ can be computed indirectly by solving linearly for the vector $[\sin(\theta/2), \cos(\theta/2)]$

[1] We can consider the ratio $L/\rho$ instead of $L$ and $\rho$ individually because the essential matrix is defined up to a scale factor.

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

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 Eq. (9), that is,

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

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 $e^* = [\sin(\theta/2), \cos(\theta/2)]$ corresponding to the minimum eigenvalue minimizes the sum of squares of the residuals, subject to $\|e^*\| = 1$. Finally, $\theta$ can be computed from $e^*$.

### 4.4. Discussion

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 up to which limit we can actually push $L$ so that our restrictive model is still usable. Indeed, as observed in the expression of the essential matrix (7), when $L \neq 0$, the model is described by two parameters ($\theta$ and $L/\rho$), that is, at least two point correspondences are required to estimate the camera motion.[3] However, as we will point out in Section 6, our 1-point parameterization continues still to be a very good approximation in those cases in which $\theta$ is small ($\theta < 10$ deg).

Also, 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 still holds very well also in the real situations; the performance will be analyzed in Section 6.

## 5. OUTLIER REMOVAL

Outlier removal is the most delicate process in camera pose estimation. The presence of 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 (six DoF) can be computed from all the remaining inliers using standard methods (Hartley & Zisserman, 2004; Stewenius et al., 2006).

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

### 5.1. 1-Point RANSAC

The first step of our 1-point RANSAC consists of computing the relative motion out of one randomly chosen correspondence. To do this, we first use Eq. (10). The motion hypothesis is then generated using Eq. (2) (note that $\rho$ can be arbitrarily set to 1). The second step is counting the inlier rate in each iteration, that is, the number of correspondences that satisfy the hypothesis. This can be done using the reprojection error.[4] We used an error threshold of one pixel. Note that for an efficient computation of the reprojection error, some approximation exists, e.g., the *Sampson* distance (Hartley & Zisserman, 2004) or the *directional error* (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, $\theta$ is computed from each feature correspondence using Eq. (10); then a histogram $H$ can be built in which each bin contains the number of features that count for the same $\theta$. A sample histogram built from real data is shown in Figure 4. When the circular motion model is well satisfied, the histogram has a very narrow peak centered on the best motion estimate $\theta^*$, that is, $\theta^* = \text{argmax}\{H\}$. We then generate our motion hypothesis by substituting $\theta^*$ into Eq. (2) and use the reprojection error to identify all the inliers.

To make the algorithm even more efficient, we should avoid the computation of the histogram. Therefore, instead of computing $\theta^*$ as the argmax of the histogram, we set $\theta^*$ 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 the median giving as good results as the argmax and therefore we used this in our final implementation. A comparison between the median and ground-truth data is shown in Figure 16 later in the paper. Notice that using the median we are completely avoiding the computation of the histogram. However, for the sake of clarity we will still refer to this approach as the histogram voting method.

Compared to 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 on average 20 $\mu$s when the average number of putative matches between the two images was about 3,000 image points. The 1-point RANSAC found a successful solution in fewer than seven

[3] Note that because $\rho$ does not appear as a multiplicative factor in Eq. (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, Fraundorfer, Pollefeys, & Siegwart, 2009).

[4] To compute the reprojection error, we used the *gold-standard method* in Hartley and Zisserman (2004).

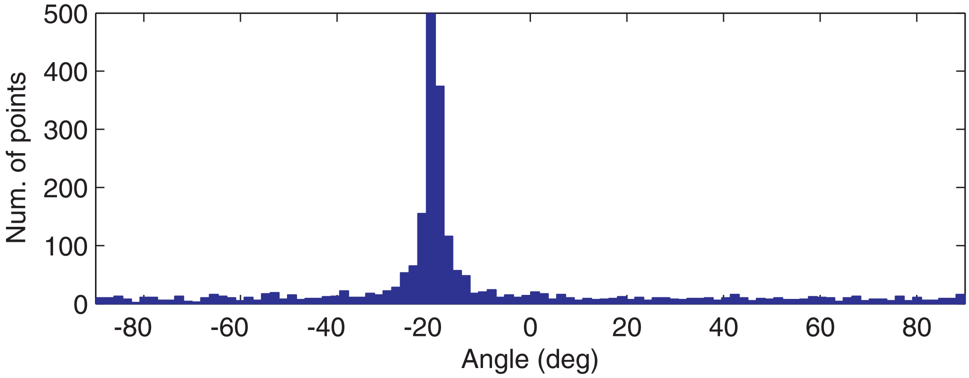
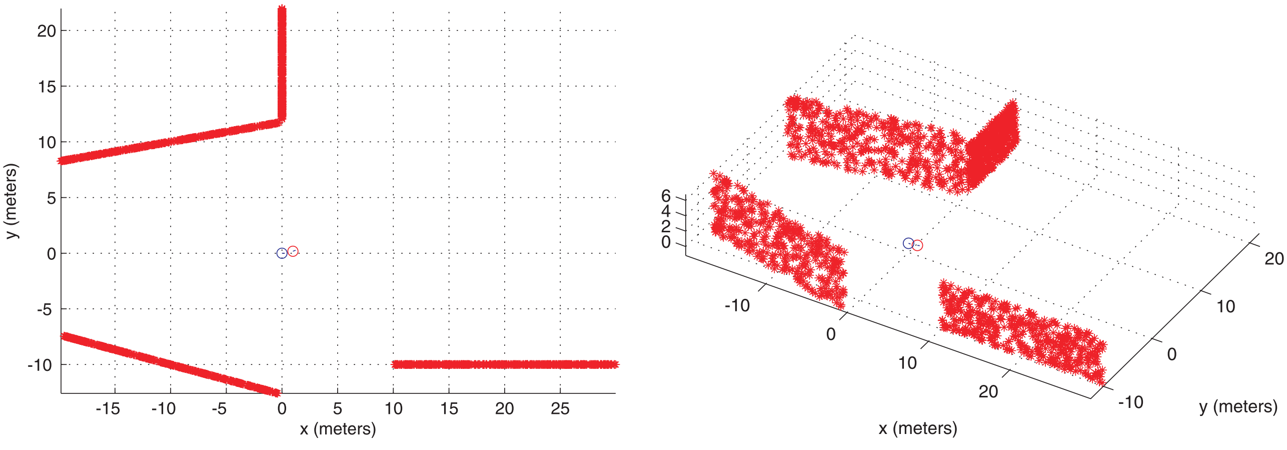


**Figure 4.** Sample histogram from feature correspondences.



(a) (b)

**Figure 5.** Our synthetic scenario: (a) top view and (b) 3D view.

iterations, requiring at most 2 ms. These tests were done with an Intel 2-GHz Dual-Core laptop.

## 6. EXPERIMENTS

In this section, we validate our motion model. The 1-point method and the histogram voting method are 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 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 Section 4.4, in order to use our 1-point parameterization the camera needs to be installed above the rear-wheel axis, thus satisfying the requirement $L = 0$. In this section, we evaluate also under which motion conditions we can arbitrarily fix the camera on the vehicle. The position of the camera is in fact of utmost importance in commercial automotive applications, in which the camera is usually under the vehicle windscreen.

We also evaluate the performance when the planarity constraint is not perfectly satisfied. For the 5-point method, we use the implementation of the algorithm available at the authors' website (Nister, 2003). We first compare the three algorithms on synthetic data and finally on real data.

### 6.1. Experiments on Synthetic Data

#### 6.1.1. 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 Figure 5. We set the first camera at the origin and randomize scene points uniformly inside several different planes, which stand for the facades of urban buildings. We used overall 1,600 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 $\theta$, on the vehicle displacement $\rho$, and on the distance $L$ of the camera from the center of the rear wheels. These parameters are the same as introduced in the preceding sections.

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 $640 \times 480$ pixel image.

#### 6.1.2. 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 algorithm (Nister, 2003; Stewenius et al., 2006). For completeness, we also added the comparison with the 2-point RANSAC (Ortin & Montiel, 2001). The performance is done by comparing the percentage of inliers found by all the 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 $\theta$ and the camera offset $L$. Because this would require us to do the test for all the possible combinations of $\theta$ and $L$, we chose to show here only two extreme cases, that is, the optimal case $L = 0$ m and the case $L = 1$ m. In fact, these two cases are those we tested also on our platform, and therefore we decided to replicate them in simulation. For larger $L$ we found that the performance decreases linearly.

The average results, over 1,000 trials, are shown in Figure 6 for planar and non–perfectly planar motion, respectively. For simulating a nonplanar motion, we introduced a 0.1-m-high step and a tilt angle of 1 deg. Note that we limited the range of $\theta$ in the simulations to be between 0 and 20 deg as this is what we experienced with the real data from our platform (see Figure 7). Note that each plot in Figure 6 corresponds to a different combination of motion (planar/nonplanar) and camera settings ($L = 0/L = 1$). For each combination, we generated 1,000 trials; each trial consists of perturbing the image points with 0.5-pixel, standard-deviation Gaussian noise. Every dot in the plot shows the median over these 1,000 trials for a given $\theta$ angle.

As observed in Figure 6(a), for planar motion and $L = 0$, the performance of the algorithms stays constant with $\theta$ as expected. However, when $L = 1$ [Figure 6(b)], the fraction of inliers found by the 1-point and histogram-voting methods decreases with $\theta$ and becomes lower than with the 5-point RANSAC at $\theta = 10$ 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 $\theta$ regardless of $L$. The 5-point method indeed does not assume motion constraints.

For non–perfectly planar motion [Figures 6(c) and 6(d)], the performance of the 1-point and histogram-voting methods decreases notably, with only 50% of the inliers detected.

For the 2-point RANSAC, the number of found inliers in general stays constant independently of $L$. The performance drops remarkably when the motion becomes nonplanar.

#### 6.1.3. Number of RANSAC Iterations

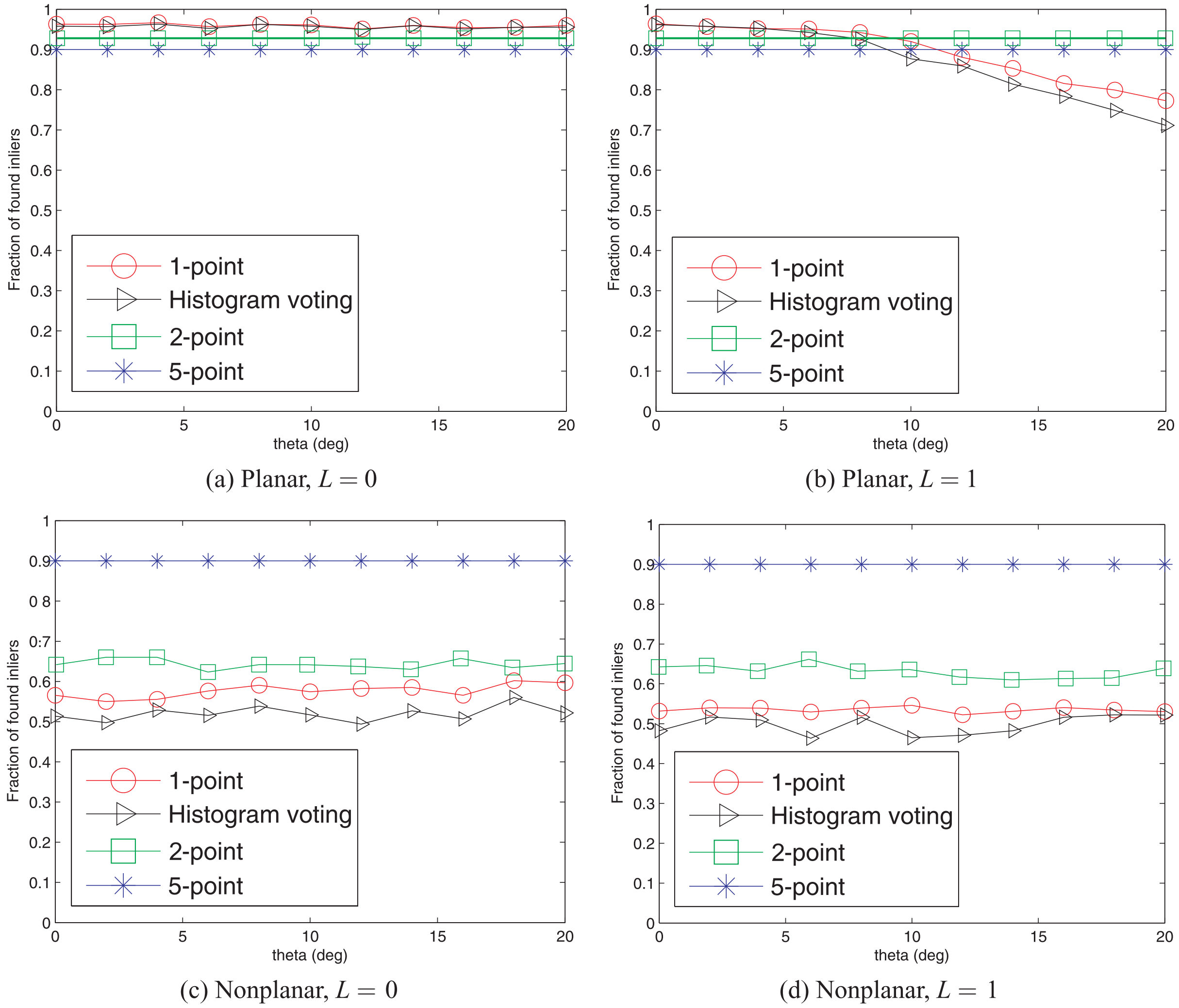
We repeated the experiments presented in the preceding section by varying also the percentage of outliers in the data points from 10% up to 90%. The results were the same as in Figure 6 regardless of the number of outliers in the data points. However, the number of RANSAC iterations needed to find the largest set of inliers increased exponentially with the percentage of outliers.[5] For instance, when the outliers were 70% of the data points, the 5-point RANSAC needed more than 1,500 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 Figure 8. These results are the average over different trials. Note that here we also added a comparison with the 2-point RANSAC.

As predicted by Eq. (1), the number of iterations of the 1-point and 5-point RANSAC increases exponentially with 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 2,000 iterations, whereas the 1-point method required only 90 iterations. The histogram voting method does not require iterations but is shown here just for comparison.

### 6.2. Experiments on Real Data

Note that 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.

[5] 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 Eq. (1).



(a) Planar, $L = 0$

(b) Planar, $L = 1$

(c) Nonplanar, $L = 0$

(d) Nonplanar, $L = 1$

**Figure 6.** Comparison between 1-point RANSAC, 2-point RANSAC, 5-point RANSAC, and histogram voting: Fraction of inliers versus $\theta$.

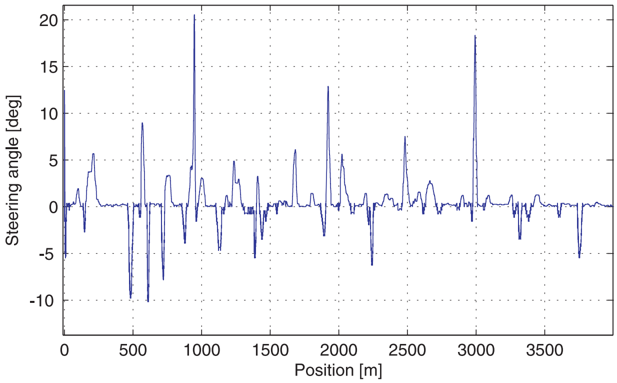


**Figure 7.** Steering angle $\theta$ (deg) versus traveled distance (m) read from our car. It is the angle the vehicle rotated between two consecutive frames.

#### 6.2.1. Data Acquisition

The method described in this paper has been successfully tested on real vehicles, both indoors and outdoors. We first present results on the outdoor vehicle in our lab. Additional results with different vehicles are given in Section 6.2.6.

The first vehicle used in our experiments is depicted in Figure 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 rear-wheel axis ($L = 0$) and camera above the windscreen ($L = 1$ m) (Figure 9). To do this, we collected two data sets 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

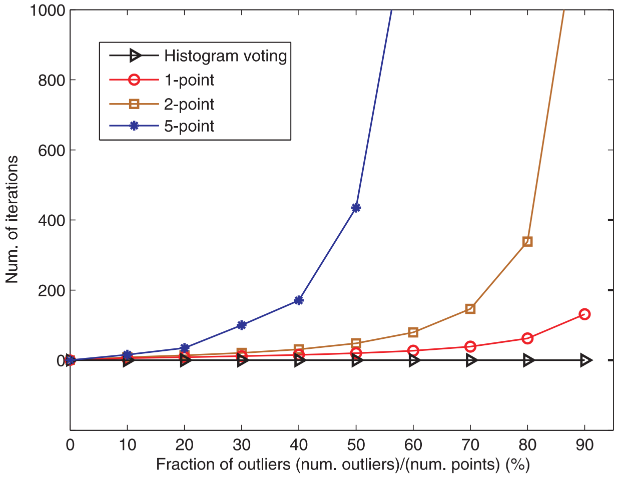


**Figure 8.** Number of RANSAC iterations versus fraction of outliers.

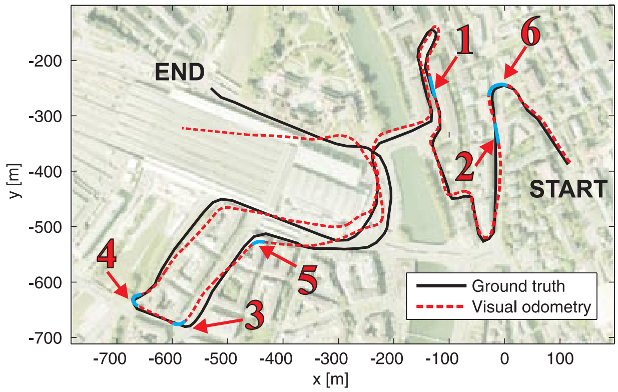


**Figure 10.** Comparison between visual odometry (red dashed line) and ground truth (black solid line). The entire trajectory is 3 km long. The numbers correspond to the sequences analyzed in Figure 11. Blue lines mark starting and ending points of each sequence.

below 10 Hz because of the memory sharing on the onboard computers. For calibrating the camera we used the toolbox described in Scaramuzza, Martinelli, and Siegwart (2006) and available from the authors' website. The vehicle speed ranged between 0 and 45 km/h.

The data set was taken in normal traffic in the city center of Zurich during a 3-km trajectory (Figure 10). Therefore, many pedestrians, moving trams, buses, and cars were also present. Point correspondences were extracted using the Harris detector (Harris & Stephens, 1988).

### 6.2.2. 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 all 4,000 images in our data set, we decided to show them only for some selected paths. The results of the comparison are presented in Figure 11, and the paths they refer to are shown in Figure 10. As observed in Figure 11, the performance of the 1-point and histogram voting methods compare very well with that of the 5-point method for the first four cases (a–d). The performance of the two algorithms is slightly minor in the fifth path [Figure 11(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 frames be small. The performance drastically decreases at some point in the sixth path where the car is going downhill on a slightly twisting road. Also

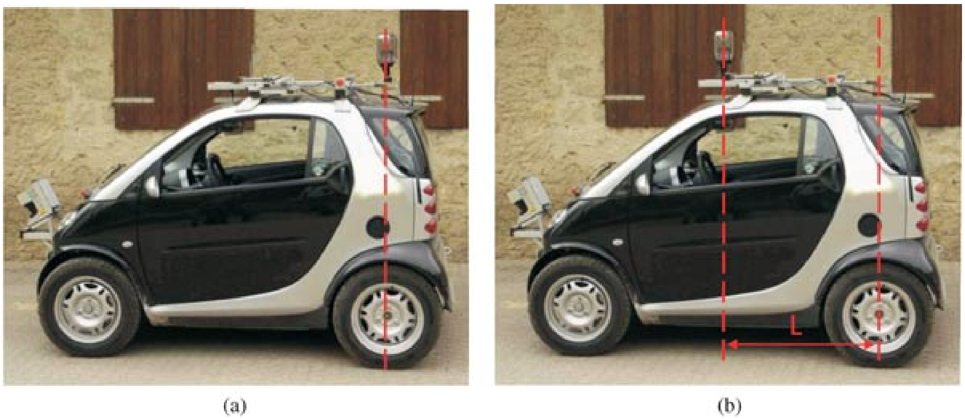
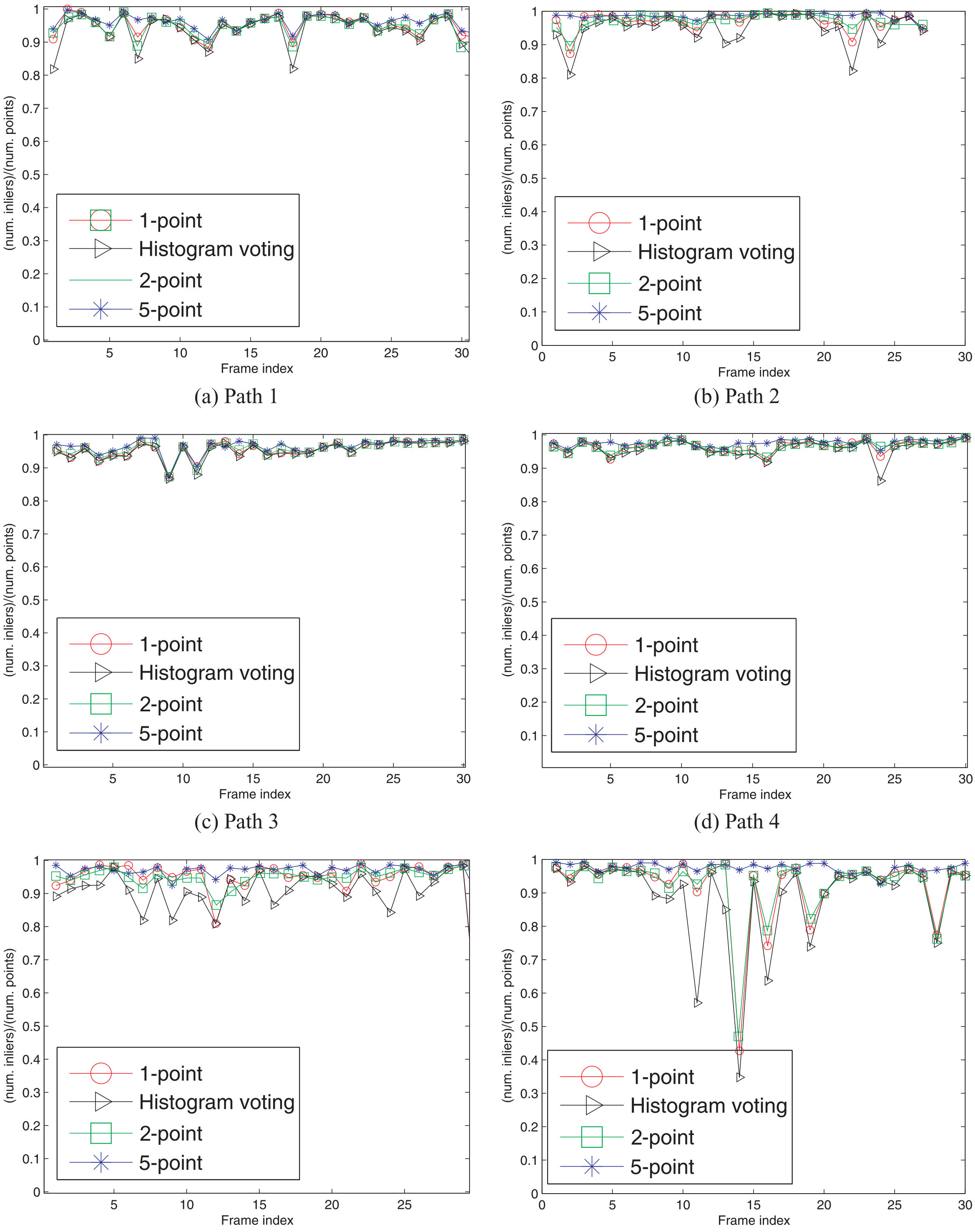


**Figure 9.** Our vehicle equipped with an omnidirectional camera. The two different settings $L = 0$ m and $L = 1$ m used in the experiments are shown.



(a) Path 1

(b) Path 2

(c) Path 3

(d) Path 4

(e) Path 5

(f) Path 6

**Figure 11.** 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 railways), $L = 1$ 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; and (f) curving path, non–perfectly flat road (going downhill with slightly twisting road), $L = 1$ m.

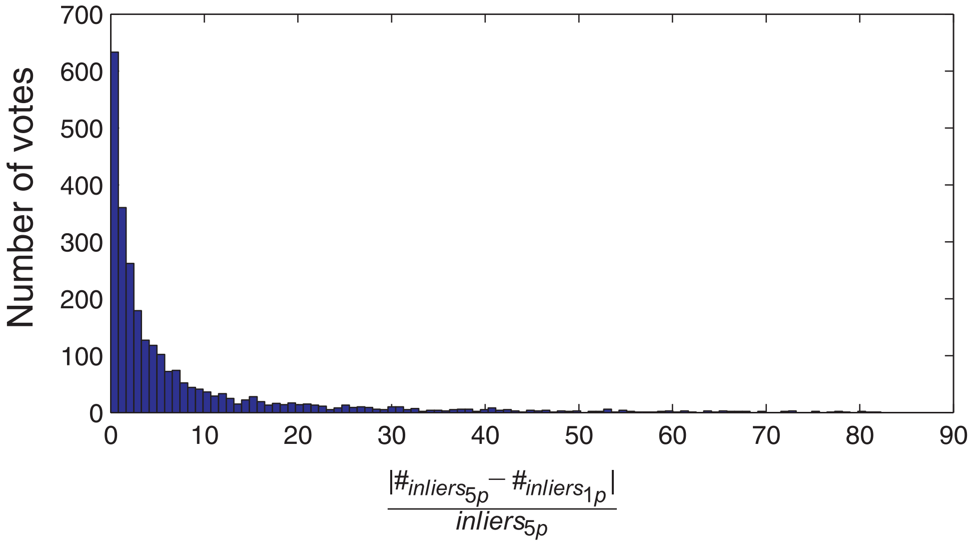


**Figure 12.** Histogram of the relative difference (%) between the inlier count of the 1-point and the 5-point algorithm over all consecutive image pairs. This difference is computed as $|\#_{\text{inliers}_{5p}} - \#_{\text{inliers}_{1p}}|/\#_{\text{inliers}_{5p}}$. As observed, the percentage of inliers of the 1-point method differs from that of the 5-point method by less than 10% in 80% of the cases. The histogram voting method gave the same performance, and therefore it is not shown here.

notice that, overall, the performance of the 2-point RANSAC is very similar to that of the 1-point and histogram voting methods. Therefore, in the following we will concentrate our comparison mainly with respect to the 5-point RANSAC.

By inspecting the performance for the entire data set, 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 Figure 12, which shows the histogram of the relative difference (percent) 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 nonperfect planarity of the road. To verify the 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 $\#_{\text{inliers}_{1p}}/\#_{\text{inliers}_{5p}}$. The relation between the nonplanarity of the estimated motion and the inlier ratio is shown in Figures 13 and 14. These plots depict the 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 Section 6.1.2.

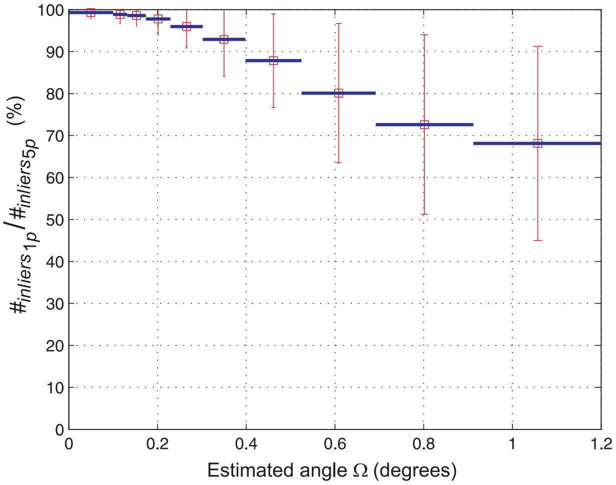


**Figure 13.** Effect of the estimated tilt angle Ω on the ratio between the inlier count of the 1-point and the inlier count of the 5-point algorithm: $(\#_{\text{inliers}_{1p}}/\#_{\text{inliers}_{5p}})$. Mean and standard deviation of this ratio are computed within predefined intervals of Ω.

Despite this, from Figure 12 we can see that our restrictive motion model is a good approximation of the motion of the car. Furthermore, in all the experiments we found that the 1-point and the histogram voting methods performed the same. However, we also observed that in the presence of low frame rate or nonplanar motion the performance of the

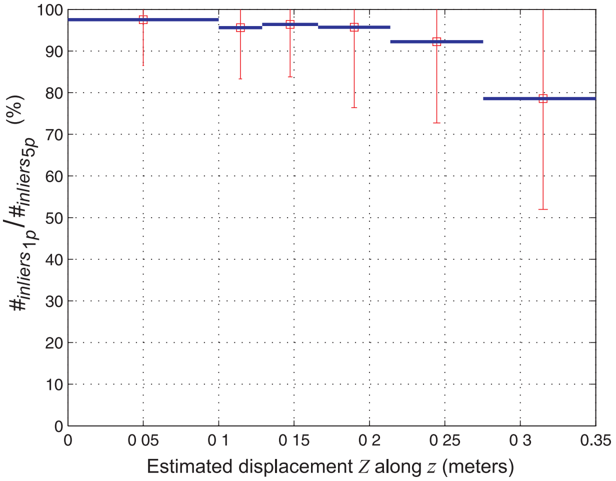


**Figure 14.** Effect of the estimated displacement $Z$ along $z$ on the ratio between the inlier count of the 1-point and the inlier count of the 5-point algorithm: ($\#_{\text{inliers}_{1p}}/\#_{\text{inliers}_{5p}}$). Mean and standard deviation of this ratio are computed within predefined intervals of $Z$.

histogram voting was slightly lower. Regarding the computational cost, during all the experiments we found that the 1-point RANSAC required at most 7 iterations, the 2-point needed about 300 iterations, and the 5-point needed from 500 up to 2,000 iterations.

### 6.2.3. Visual Odometry

To evaluate the quality of point correspondences output by our proposed methods, we implemented a motion estimation algorithm and we ran it on the entire 3-km data set. For this experiment, we implemented a very simple, incremental motion estimation algorithm, which means that we computed the motion only between consecutive frames (e.g., two-view SFM). Note that we did not use the previous poses and structure to refine the current estimate. Furthermore, we did not use bundle adjustment.

For feature detection we used the Harris detector (Harris & Stephens, 1988), whereas for feature matching we used the sum of squared differences. The maximum initial number of features was about 3,000—before removing the inliers—and between 1,000 and 2,000 after removing the outliers.

For removing the outliers, we used one of our proposed methods. From the remaining inliers, the relative pose was then estimated using the 5-point algorithm[6] from Stewenius et al. (2006), which provides unconstrained five-DoF motion estimates.[7] 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 Figure 10 overlaid on a satellite image. Note that this algorithm runs at 400 frames per second (fps).[8]

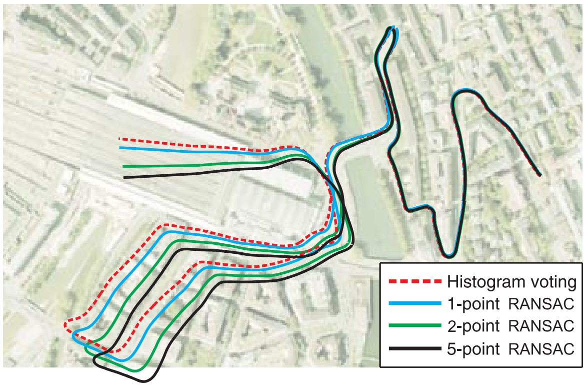


**Figure 15.** Comparison between visual odometry trajectories using the three different methods for outlier removal: histogram voting (red dashed line), 1-point RANSAC (cyan solid line), 2-point RANSAC (green solid line), and 5-point RANSAC (black solid line).

Figure 15 shows instead the comparison among the visual odometry paths computed with histogram voting and 1-point, 2-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 outperform both the 2-point and the 5-point RANSAC. This result should not surprise the reader. Indeed, let us recall that we did not use bundle adjustment, which obviously would largely reduce the accumulated drift.[9] However, it is also important to point out that sometimes the found inliers are not the largest RANSAC, meaning that more iterations would have actually been necessary. Additionally, this result points out that even though for most of the frames the 2-point and the 5-point RANSAC find a few more inliers than the 1-point RANSAC, the 1-point RANSAC and the histogram voting methods output “better” inliers in that they favor the underlying motion model.

[6] Note that the 5-point algorithm is not the 5-point RANSAC. The 5-point algorithm optimizes the five-DoF pose of the vehicle in the least-squares sense. It is called “5-point” because it requires a minimum of five points to work, although it can actually be applied to $n$ points.

[7] Because translation is recovered up to scale, we have five DoF.

[8] Please note that a demonstrative video of the resulting online trajectory estimation can be watched at http://www.youtube.com/watch?v=t7uKWZtUjCE.

[9] The accumulated drift in this 3-km data set was about 30 m in $x-y$ and 1 m in $z$.

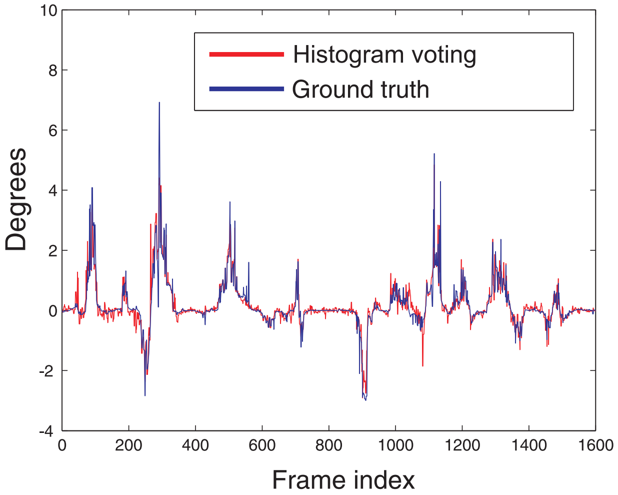


**Figure 16.** Comparison between the yaw angle ($\theta$) and ground truth. In this case the yaw angle was computed with the histogram voting method (that is, the median of the distribution). The ground truth was recorded from an IMU and an optical gyroscope.

### 6.2.4. Orientation Error: Comparison with Ground-Truth Data

For ground-truth data acquisition, the vehicle was equipped with an inertial measurement unit (IMU) and an optical gyroscope. The fusion of the orientation from these two sources was done by means of an EKF as described in Lamon, Kolski, and Siegwart (2006).

A comparison between the yaw angle ($\theta$) and the ground truth for the first 1,600 frames is depicted in Figure 16.

In Figure 17, we compare the error in orientation between the histogram voting and the 5-point RANSAC with respect to the ground truth for the first 1,000 frames . The error is calculated for the roll, pitch, and yaw angles. First, we can notice that the errors of the two algorithms are very well correlated. Additionally, they look very similar. However, notice that the yaw error introduced by the 5-point RANSAC is larger than that of the histogram voting method on a few occasions. As we mentioned at the end of the preceding section, this can be explained by the fact that the 1-point motion parameterization captures the dominant components of the motion, which is mostly planar and circular.

### 6.2.5. Coping with Few Features and Occlusions

Notice that this is a very challenging data set. Sometimes features appeared only in some spots of the image (e.g., on the road when crossing the bridge), and therefore in these situations the result of the 5-point algorithm introduced large errors that depend on the position of the features in the image. Conversely, the 1-point algorithm can cope with

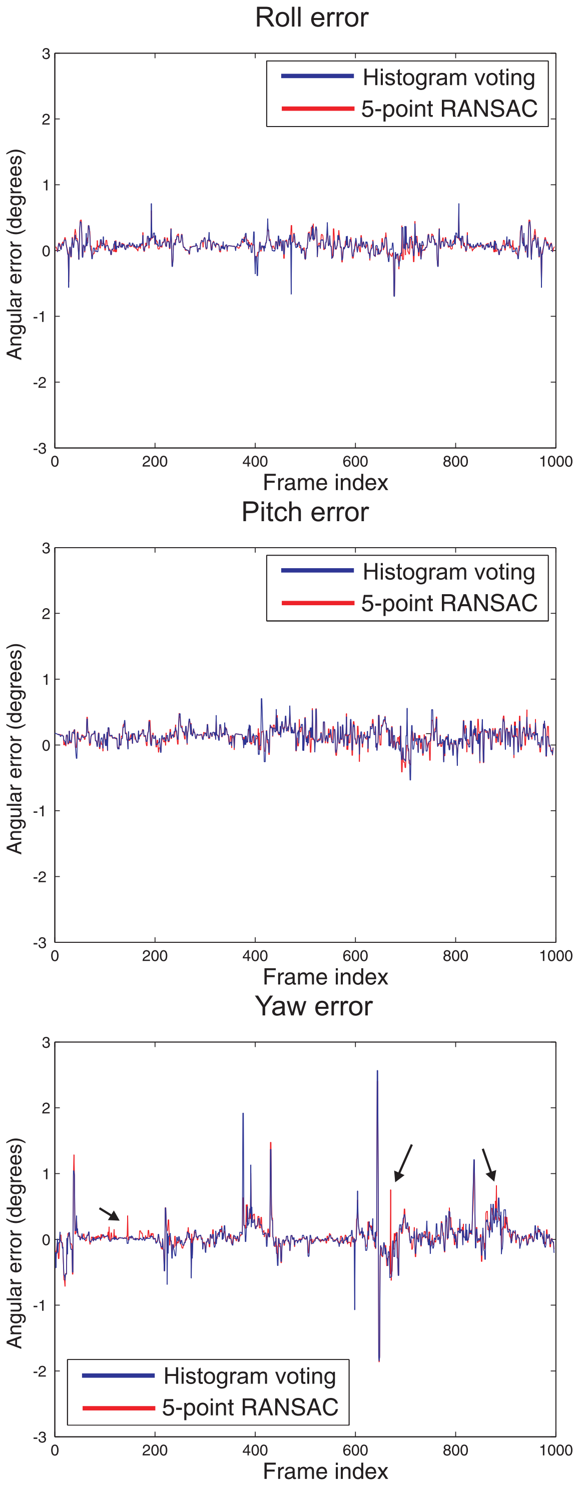
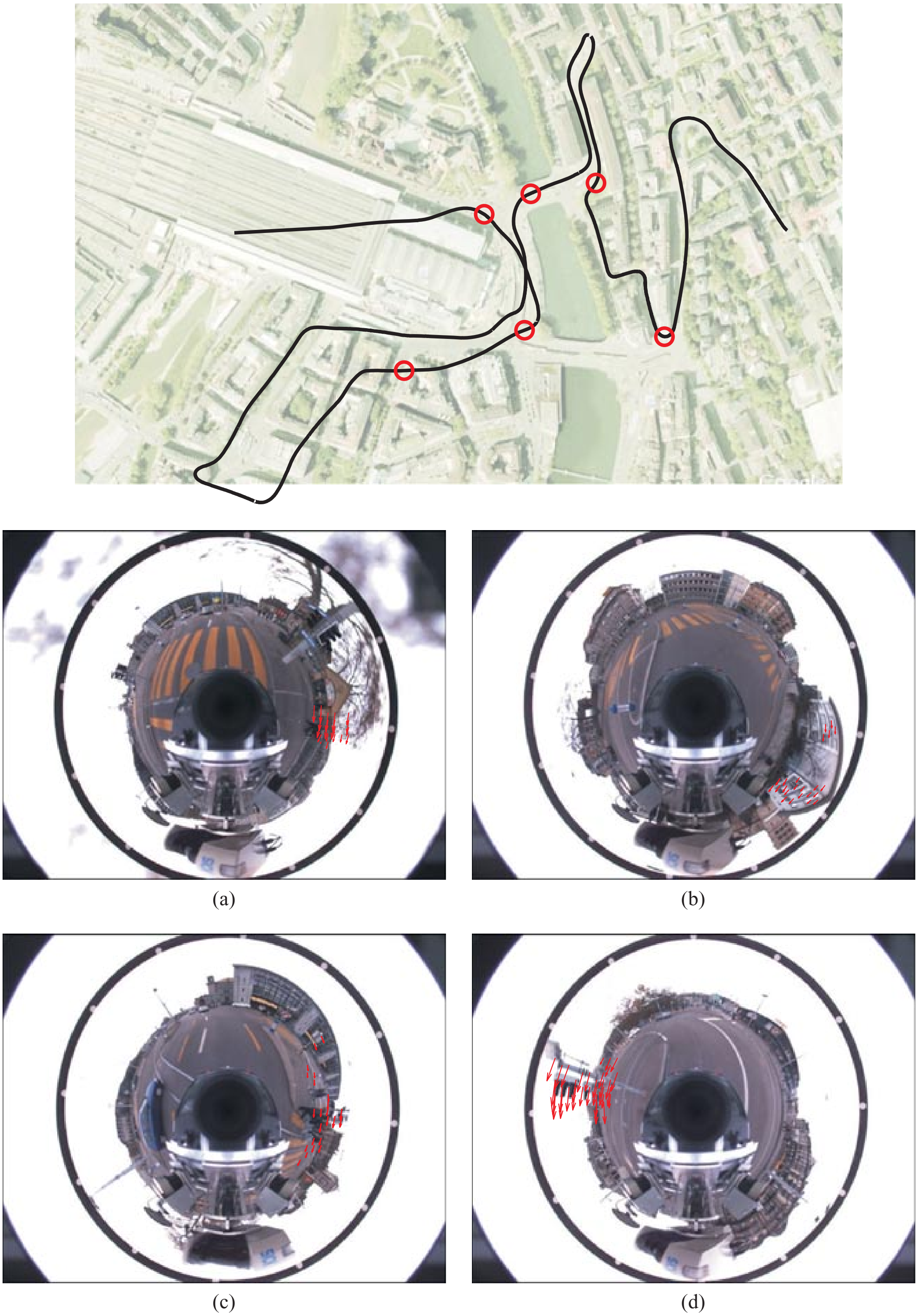


**Figure 17.** Comparison between estimated orientations: Blue, Histogram voting; red, 5-point RANSAC. The arrows indicate spots where the error of the 5-point RANSAC was larger than that of the histogram voting. This happened, however, only a few times over 1,000 frames.


(a)

(b)

(c)

(d)

**Figure 18.** Top, Trajectory estimated by the 5-point RANSAC with highlighted spots where the 5-point RANSAC failed. (a–d) Some sample images taken at some of those spots with overlaid feature matches.

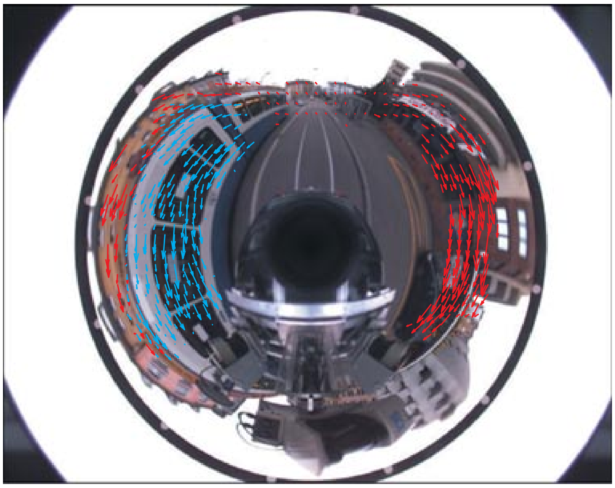


Figure 19. Large occlusion by a passing tram. The arrows show the inliers found by either algorithm: Blue, 5-point RANSAC; red, 1-point RANSAC. The two ends of each arrow indicate the positions of the feature in the first and in the second image.

very few features due to the implicit motion constraint. In Figure 18 we replotted the trajectory estimated by the 5-point RANSAC. The six circles identify spots where only a few inliers were found. In these cases the motion estimate of the 5-point algorithm was completely wrong, with errors in rotation estimation about 20 deg off. This was caused by the fact that we had only very few feature points and that they were concentrated in small spots of the image [see Figures 18(a)–18(d)]. To overcome these problems, when they happened we replaced the motion estimate of the 5-point algorithm with that of the 1-point algorithm [expression (11) and Eq. (12)]. These situations were easy to detect: we just checked that the difference of the rotation estimate output by the 5-point and 1-point algorithms did not exceed 10 deg.

Another interesting situation that sometimes occurred was the occlusion of large parts of the camera field of view by other moving vehicles. In Figure 19 we show an interesting situation in which a tram is driving on the opposite side of the road. As the reader can observe, about half of the left side of the camera image remains occluded by the tram. The inliers found by the 5-point RANSAC and by the 1-point RANSAC are drawn in blue and red, respectively. Notice that the inliers output by the 1-point RANSAC come from static objects, whereas those output by the 5-point RANSAC come from the moving tram. The reason is the following. The relative motion of the tram with respect to the center of our vehicle is of course locally planar and circular, but the instantaneous center of rotation of this combined motion does not lie along the axis of the rear wheels as the 1-point RANSAC would expect it to be in the ideal case. This huge difference makes our 1-point RANSAC *believe* that the inliers are those coming from the static objects. Conversely, the 5-point RANSAC—which does not use a motion model—selects the largest set of inliers, which in this case are the feature points on the tram. This fact is very remarkable and should convince the reader of the importance of exploiting motion models in visual odometry.

### 6.2.6. Visual Odometry Results on Different Data Sets

We ran our visual odometry approach on other data sets acquired with different vehicles, both outdoors and indoors. The first data set was collected with the vehicle depicted

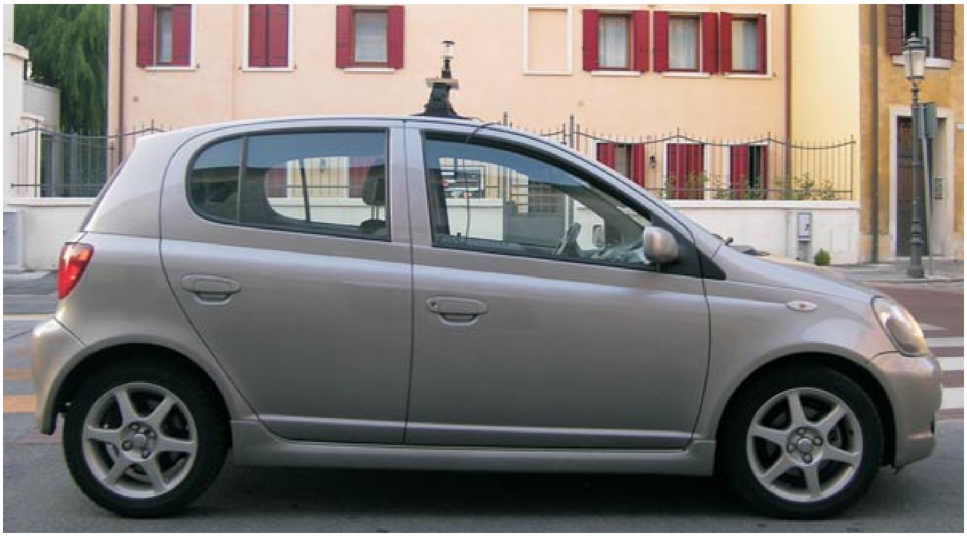


Figure 20. The vehicle and camera used to capture the Padua data set kindly provided by A. Pretto (Pretto et al., 2011).

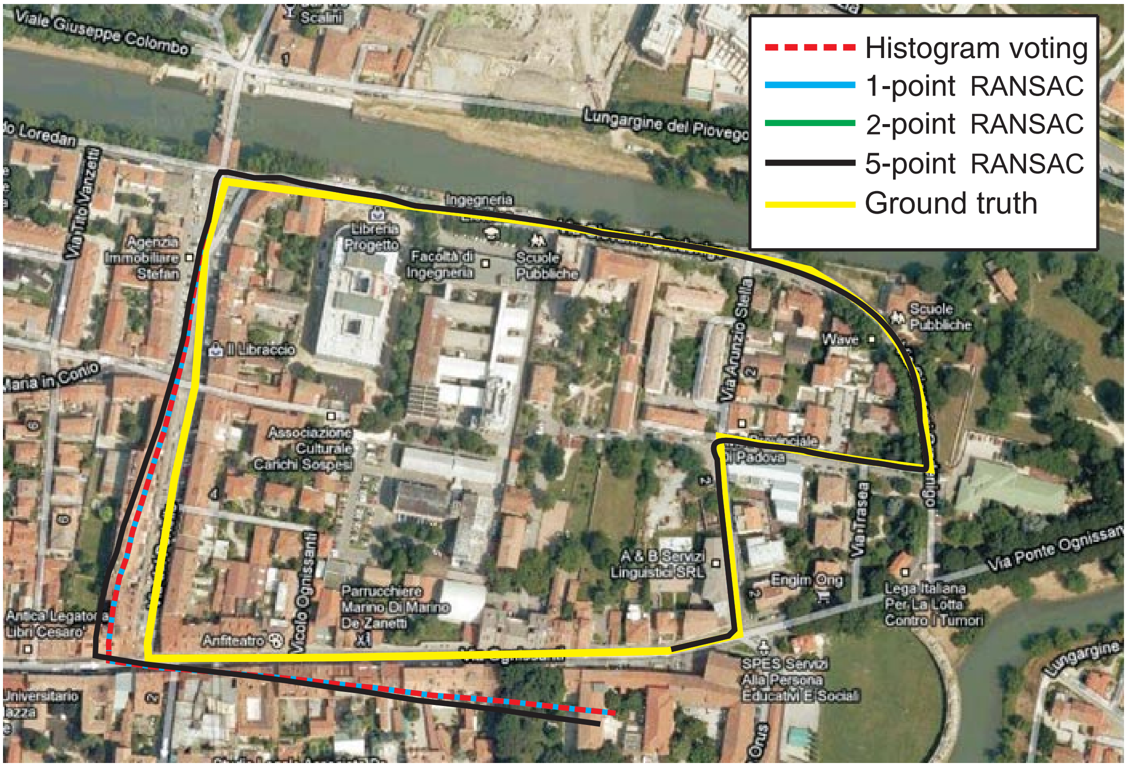


**Figure 21.** Comparison between visual odometry trajectories for the Padua data set (Pretto et al., 2011) using the three different methods for outlier removal: Histogram voting (red dashed line), 1-point RANSAC (cyan solid line), 2-point RANSAC (green solid line), and 5-point RANSAC (black solid line). The length of the trajectory was 1.4 km. Notice that the trajectory estimated by the 2-point RANSAC is not visible because it coincides almost perfectly with that of the 1-point and histogram voting methods.

in Figure 20. The data set was kindly provided by Pretto, Menegatti, and Pagello (2011). The camera was a Basler Scout—resolution of 1,032 × 778 pixels—equipped with a hyperbolic mirror. The data set was collected in the city of Padua. The reconstructed trajectories using the methods described in the preceding sections are shown in Figure 21. As can be observed, again the performance of the 1-point RANSAC is slightly better than that of the 2-point and the 5-point RANSAC.

For the indoor experiment, we used the differential-drive robot depicted in Figure 22. The resulting trajectories estimated by the different methods are shown in Figure 23. As can be observed, they overlay almost perfectly on each other. This is because the different algorithms output practically the same inliers.

## 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 results 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 when the camera is positioned above the rear-wheel axis ($L = 0$). However, in the experimental section we have shown that also for the case $L \neq 0$ our restrictive model is still usable under the condition that the rotation angle $\theta$ between two camera poses is small (or alternatively that the frame rate is high). In particular we have shown that in most cases our 1-point RANSAC and histogram voting method perform as well as or even better than the standard 5-point RANSAC, finding almost the same number of inliers. The better performance with respect to the 5-point RANSAC can be explained by the fact that the 1-point RANSAC and histogram voting methods find "better" inliers in that they capture the dominant component of the motion of the vehicle, which is mostly planar and circular. Finally, we showed the quality of the output correspondences by comparing the estimated motion and orientation of the car with ground-truth data.

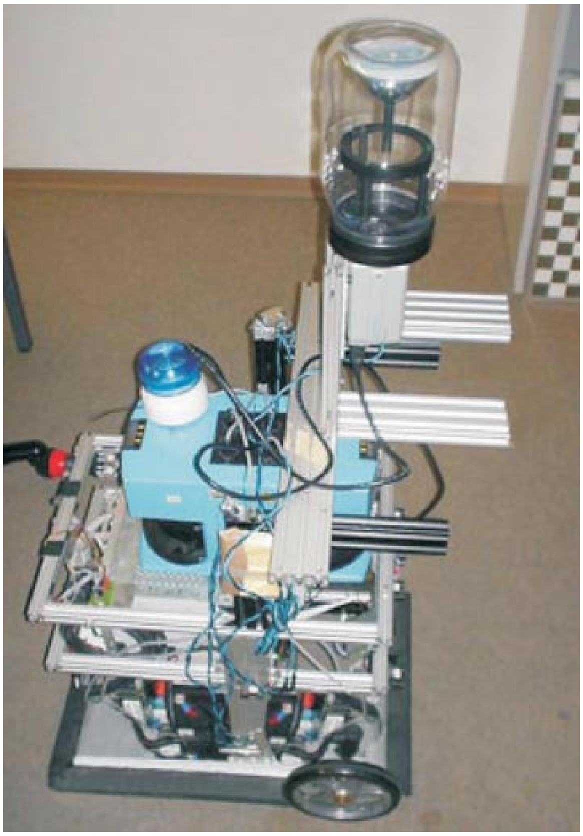


**Figure 22.** The differential-drive robot and the camera used to capture the indoor data set. Notice that the camera was placed intentionally above the wheel axis to make the 1-point motion parameterization valid.

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 satisfied, this is reflected in a reduced number of inliers found by the 1-point RANSAC and histogram voting method. However, when this happens the problem can be easily overcome by switching back 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 kurtosis of the distribution provides an easy way to switch between the 1-point and 5-point approaches.

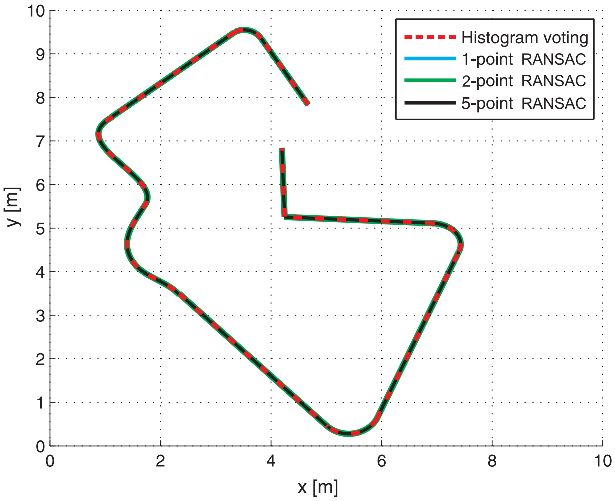


**Figure 23.** Comparison between visual odometry trajectories for the indoor data set. Because the ground truth was not available, we just compare the trajectories output by the different algorithms. Notice that the estimated trajectories overlay almost perfectly. This is because the different algorithms output practically the same inliers.

## ACKNOWLEDGMENTS

The author thanks Dr. Alberto Pretto for providing the Padua data set and all five reviewers of this paper for their useful comments and suggestions.

## REFERENCES

Chum, O., & Matas, J. (2005, June). Matching with PROSAC—Progressive sample consensus. In International Conference on Computer Vision and Pattern Recognition, San Diego, CA.

Civera, J., Grasa, O., Davison, A., & Montiel, J. (2010). 1-Point RANSAC for EKF filtering: Application to real-time structure from motion and visual odometry. Journal of Field Robotics, 27, 609–631.

Clemente, L. A., Davison, A. J., Reid, I., Neira, J., & Tardos, J. D. (2007, June). Mapping large loops with a single hand-held camera. In Robotics Science and Systems, Atlanta, GA.

Corke, P. I., Strelow, D., & Singh, S. (2005, August). Omnidirectional visual odometry for a planetary rover. In IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Alberta, Canada.

Davison, A. (2003, October). Real-time simultaneous localisation & mapping with a single camera. In International Conference on Computer Vision, Nice, France.

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

Eade, E., & Drummond, T. (2007, October). Monocular SLAM as a graph of coalesced observations. In IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil.

Faugeras, O., & Maybank, S. (1990). Motion from point matches: Multiplicity of solutions. International Journal of Computer Vision, 4(3), 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, June). Visual vehicle egomotion estimation using the Fourier-Mellin transform. In IEEE Intelligent Vehicles Symposium, Istanbul, Turkey.

Handa, A., Chli, M., Strasdat, H., & Davison, A. J. (2010, June). Scalable active matching. In International Conference on Computer Vision and Pattern Recognition, San Francisco, CA.

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, UK: Cambridge University Press.

Jung, I., & Lacroix, S. (2005, October). Simultaneous localization and mapping with stereovision. In Robotics Research: The 11th International Symposium, San Francisco, CA.

Klein, G., & Murray, D. (2008, December). Improving the agility of keyframe-based SLAM. In European Conference on Computer Vision, Marseille, France.

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

Lacroix, S., Mallet, A., Chatila, R., & Gallo, L. (1999, June). Rover self localization in planetary-like environments. In International Symposium on Artificial Intelligence, Robotics, and Automation for Space (i-SAIRAS), Noordwijk, The Netherlands (pp. 433–440).

Lamon, P., Kolski, S., & Siegwart, R. (2006, September). The SmartTer—A vehicle for fully autonomous navigation and mapping in outdoor environments. In International Conference on Climbing and Walking Robots, Brussels, Belgium.

Lemaire, T., & Lacroix, S. (2007). SLAM with panoramic vision. Journal of Field Robotics, 24(1–2), 91–111.

Lhuillier, M. (2005, October). Automatic structure and motion using a catadioptric camera. In IEEE Workshop on Omnidirectional Vision, Beijing, China.

Longuet-Higgins, H. (1981). A computer algorithm for reconstructing a scene from two projections. Nature, 293(10), 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(3), 169–186.

Milford, M., Wyeth, G., & Prasser, D. (2004, April). RatSLAM: A hippocampal model for simultaneous localization and mapping. In IEEE International Conference on Robotics and Automation (ICRA'04), New Orleans, LA.

Milford, M. J., & Wyeth, G. (2008, May). Single camera vision-only SLAM on a suburban road network. In IEEE International Conference on Robotics and Automation (ICRA'08), San Francisco, CA.

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

Nister, D. (2003, June). An efficient solution to the five-point relative pose problem. In International Conference on Computer Vision and Pattern Recognition, Madison, WI.

Nister, D. (2005). Preemptive RANSAC for live structure and motion estimation. Machine Vision and Applications, 16(5), 321–329.

Nister, D., Naroditsky, O., & Bergen, J. (2006). Visual odometry for ground vehicle applications. Journal of Field Robotics, 23(1), 3–20.

Oliensis, J. (2002). Exact two-image structure from motion. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(12), 1618–1633.

Ortin, D., & Montiel, J. M. M. (2001). Indoor robot motion based on monocular images. Robotica, 19(3), 331–342.

Philip, J. (1996). A non-iterative algorithm for determining all essential matrices corresponding to five point pairs. Photogrammetric Record, 15(88), 589–599.

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

Pretto, A., Menegatti, E., & Pagello, E. (2011, May). Omnidirectional dense large-scale mapping and navigation based on meaningful triangulation. In IEEE International Conference on Robotics and Automation, Shanghai, China.

Raguram, R., Frahm, J., & Pollefeys, M. (2009, September). Exploiting uncertainty in random sample consensus. In International Conference on Computer Vision, Kyoto, Japan.

Scaramuzza, D. (2011). 1-Point-RANSAC structure from motion for vehicle-mounted cameras by exploiting nonholonomic constraints. International Journal of Computer Vision, 95(1), 75–85.

Scaramuzza, D., Fraundorfer, F., Pollefeys, M., & Siegwart, R. (2008, October). Closing the loop in appearance-guided structure-from-motion for omnidirectional cameras. In Eighth Workshop on Omnidirectional Vision (OMNIVIS08), Marseille, France.

Scaramuzza, D., Fraundorfer, F., Pollefeys, M., & Siegwart, R. (2009, September). Absolute scale in structure from motion from a single vehicle mounted camera by exploiting nonholonomic constraints. In International Conference on Computer Vision, Kyoto, Japan.

Scaramuzza, D., Fraundorfer, F., & Siegwart, R. (2009, May). Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC. In IEEE International Conference on Robotics and Automation (ICRA'09), Kobe, Japan.

Scaramuzza, D., Martinelli, A., & Siegwart, R. (2006, October). A toolbox for easy calibrating omnidirectional cameras. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006), Beijing, China.

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(5), 1015–1026.

Siegwart, R., Nourbakhsh, I., & Scaramuzza, D. (2004). Introduction to autonomous mobile robots, 2nd ed. Cambridge, MA: 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, September). Monocular visual odometry in urban environments using an omnidirectional camera. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'08), Nice, France.