Chapter 2
SLAM Front-End

In this Chapter we discuss our choice of front-end for SLAM, the part in charge of processing the sensor information to generate the observations that will be fed to the estimation machinery. In the context of this work, observations come in the form of relative-motion constraints between any two robot poses. They are typically obtained with the Iterative Closest Point (ICP) algorithm [1] when working with laser scans. When using stereo images, the egomotion of the robot can be estimated with visual odometry [2, 3]. The latter method is adopted in our contribution presented in [4] and in this Chapter we describe it in more detail and extend it with a technique to model the uncertainty of the relative-motion constraints.

Assuming we have a pair of stereo images acquired with two calibrated cameras fixed to the robot's frame, our approach iterates as follows: SIFT image features [5] are extracted from the four images and matched between them. The resulting point correspondences are used for least-squares stereo reconstruction. Next, matching of these 3D features in the two consecutive frames is used to compute a least-squares best-fit pose transformation, rejecting outliers via RANSAC [6].

However, the outcome of this approach is also prone to errors. Errors in locating the image features lead to errors in the location of the 3D feature points after stereo reconstruction, which eventually cause errors in the motion estimate. Modeling such error propagation allows to compute motion estimates with the appropriate uncertainty bounds. In this Chapter we introduce a technique to compute the covariance of the relative pose measurement by first-order error propagation [7].

These camera pose constraints are eventually used as relative pose measurements in the SLAM we employ in this work. They are used either as odometry measurements, when matching stereo images from consecutive poses in time, or as loop closure constraints, when computing the relative motion of the last pose with respect to any previous pose. This will be discussed in Chap. 3.

R. Valencia and J. Andrade-Cetto, *Mapping, Planning and Exploration with Pose SLAM*, Springer Tracts in Advanced Robotics 119,
DOI 10.1007/978-3-319-60603-3_2

The rest of this Chapter is structured as follows. Section 2.1 explains the feature extraction and the stereo reconstruction process. Next, the pose estimation step is shown in Sect. 2.2. Then, in Sect. 2.3 we introduce a technique to model the uncertainty of the relative motion measurement. Finally, in Sect. 2.4 we provide bibliographical notes.

2.1 Feature Extraction and Stereo Reconstruction

Simple correlation-based features, such as Harris corners [8] or Shi and Tomasi features [9], are of common use in vision-based SFM and SLAM; from the early uses of Harris himself to the popular work of Davison [10]. This kind of features can be robustly tracked when camera displacement is small and are tailored to real-time applications. However, given their sensitivity to scale, their matching is prone to fail under larger camera motions; less to say for loop-closing hypotheses testing. Given their scale and local affine invariance properties, we opt to use SIFTs instead [5, 11], as they constitute a better option for matching visual features from significantly different vantage points.

In our system, features are extracted and matched with previous image pairs. Then, from the surviving features, we compute the imaged 3D scene points as follows.

Assumming two stereo-calibrated cameras and a pin-hole camera model [12], with the left camera as the reference of the stereo system, the following expressions relate a 3D scene point $\mathbf{p}$ to the corresponding points $\mathbf{m} = [u, v]^\top$ in the left, and $\mathbf{m}' = \left[u', v'\right]^\top$ in the right camera image planes

$$\begin{bmatrix} \mathbf{m} \\ s \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_o & 0 \\ 0 & \alpha_v & v_o & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{I}_3 & 0 \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}, \tag{2.1}$$

$$\begin{bmatrix} \mathbf{m}' \\ s' \end{bmatrix} = \begin{bmatrix} \alpha'_u & 0 & u'_o & 0 \\ 0 & \alpha'_v & v'_o & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}, \tag{2.2}$$

where α_u and α_v are the pixel focal lengths in the x and y directions for the left camera, and α'_u and α'_v for the right camera, (u_o, v_o) and (u'_o, v'_o) are the left and right camera image centers, respectively. The homogeneous transformation from the right camera frame to the reference frame of the stereo system is represented by the rotation matrix $\mathbf{R}$ and translation vector $\mathbf{t} = [t_x, t_y, t_z]^\top$. $[\mathbf{m}^\top, s]^\top$ and $[\mathbf{m}'^\top, s']^\top$ are the left and right image points in homogeneous coordinates, with scale s and s', respectively, and $\mathbf{I}_3$ is a 3×3 identity matrix.

Equations 2.1 and 2.2 define the following overdetermined system of equations

$$\begin{bmatrix} (u' - u'_o)\mathbf{r}_3^\top - \alpha'_u \mathbf{r}_1^\top \\ (v' - v'_o)\mathbf{r}_3^\top - \alpha'_v \mathbf{r}_2^\top \\ -\alpha_u, 0, u - u_o \\ 0, -\alpha_v, v - v_o \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} (u'_o - u')t_z + \alpha'_u t_x \\ (v'_o - v')t_z + \alpha'_v t_y \\ 0 \\ 0 \end{bmatrix}$$
$$\mathbf{A}\,\mathbf{p} = \mathbf{b}, \tag{2.3}$$

where $\mathbf{R}$ is expressed by its row elements

$$\mathbf{R} = \begin{bmatrix} \mathbf{r}_1^\top \\ \mathbf{r}_2^\top \\ \mathbf{r}_3^\top \end{bmatrix}.$$

Solving for $\mathbf{p}$ in Eq. 2.3 gives the sought 3D coordinates of the imaged points $\mathbf{m}$ and $\mathbf{m}'$. Performing this process for each pair of matching feature in a pair of stereo images results in two sets of 3D points, or 3D point clouds, i.e. $\left\{\mathbf{p}_1^{(i)}\right\}$ and $\left\{\mathbf{p}_2^{(i)}\right\}$.

2.2 Pose Estimation

Next, we present two alternatives to compute the relative motion of the camera from two stereo images by solving the 3D to 3D pose estimation problem.

The general solution to this problem consists of finding the rotation matrix $\mathbf{R}$ and translation vector $\mathbf{t}$ that minimize the squared L_2-norm for all points in the two aforementioned clouds,

$$\left\{\hat{\mathbf{R}}, \hat{\mathbf{t}}\right\} = \underset{\mathbf{R},\mathbf{t}}{\operatorname{argmin}} \sum_{i=1}^{N} \left\| \mathbf{p}_1^{(i)} - \left(\mathbf{R}\mathbf{p}_2^{(i)} + \mathbf{t}\right) \right\|^2, \tag{2.4}$$

with N the number of points in each cloud.

For both methods, we resort to the use of RANSAC [6] to eliminate outliers. It might be the case that SIFT matches occur on areas of the scene that experienced motion during the acquisition of the two image stereo pairs. For example, an interest point might appear at an acute angle of a tree leaf shadow, or on a person walking in front of the robot. The corresponding matching 3D points will not represent good fits to the camera motion model, and might introduce large bias to our least squares pose error minimization. The use of such a robust model fitting technique allows us to preserve the largest number of point matches that at the same time minimize the square sum of the residuals, as shown in Fig. 2.1.

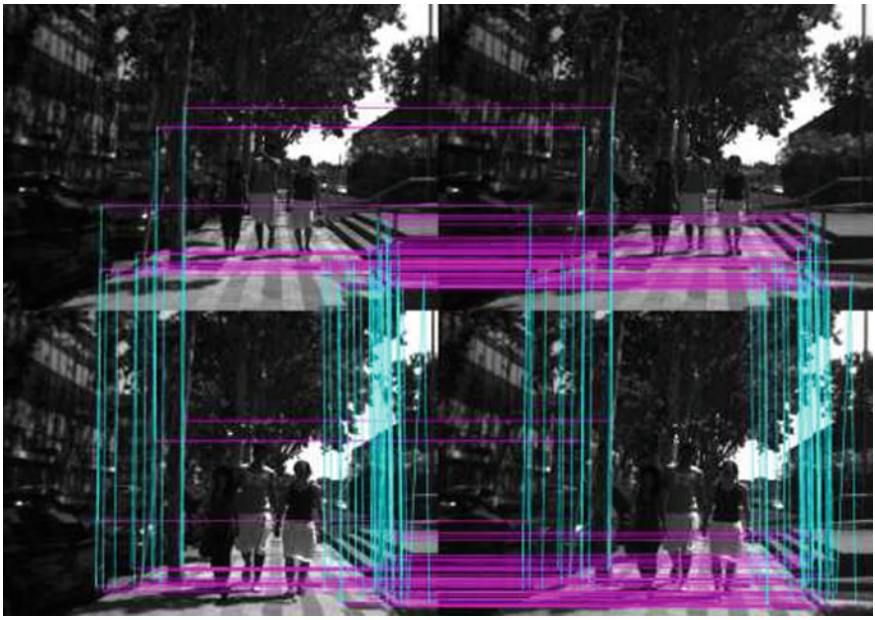


Fig. 2.1 SIFT correspondences in two consecutive stereo image pairs after outlier removal using RANSAC

Furthermore, if the covariance of the matching points is available it can be exploited so as to explicitly model their precision according to their distance from the camera. For instance, we can weight the point mismatch in Eq. 2.4 with the covariance of the triangulation of the two points. However, this would complicate further the optimization problem defined by Eq. 2.4. Instead, we chose to rely on standard techniques such as the following solutions.

2.2.1 Horn's Method

A solution for the rotation matrix $\mathbf{R}$ is computed by minimizing the sum of the squared errors between the rotated directional vectors of feature matches for the two robot poses [13]. Directional vectors $\boldsymbol{\nu}$ are computed as the unit norm direction along the imaged 3D scene point $\mathbf{p}$ and indicates the orientation of such a point, that is,

$$\boldsymbol{\nu}_1^{(i)} = \frac{\mathbf{p}_1^{(i)}}{\|\mathbf{p}_1^{(i)}\|} \tag{2.5}$$

and

$$\boldsymbol{\nu}_2^{(i)} = \frac{\mathbf{p}_2^{(i)}}{\|\mathbf{p}_2^{(i)}\|} \tag{2.6}$$

are the directional vectors for the i-th point on the first and the second point cloud, respectively.

The solution to this minimization problem gives an estimate of the orientation of one cloud of points with respect to the other, and can be expressed in quaternion form as

$$\frac{\partial}{\partial \mathbf{R}} \left(\mathbf{q}^\top \mathbf{B} \mathbf{q} \right) = 0 \,, \tag{2.7}$$

where $\mathbf{B}$ is given by

$$\mathbf{B} = \sum_{i=1}^{N} \mathbf{B}_i \mathbf{B}_i^\top \,, \tag{2.8}$$

$$\mathbf{B}_i = \begin{bmatrix} 0 & -c_x^{(i)} & -c_y^{(i)} & -c_z^{(i)} \\ c_x^{(i)} & 0 & b_z^{(i)} & -b_y^{(i)} \\ c_y^{(i)} & -b_z^{(i)} & 0 & b_x^{(i)} \\ c_z^{(i)} & b_y^{(i)} & -b_x^{(i)} & 0 \end{bmatrix} \,, \tag{2.9}$$

and

$$\mathbf{b}^{(i)} = \boldsymbol{\nu}_2^{(i)} + \boldsymbol{\nu}_1^{(i)}, \quad \mathbf{c}^{(i)} = \boldsymbol{\nu}_2^{(i)} - \boldsymbol{\nu}_1^{(i)} \quad . \tag{2.10}$$

The quaternion $\mathbf{q}$ that minimizes the argument of the derivative operator in Eq. 2.7 is the smallest eigenvector of the matrix $\mathbf{B}$.

If we denote this smallest eigenvector by the 4-tuple $(q_1, q_2, q_3, q_4)^\top$, it follows that the angle θ associated with the rotational transform is given by

$$\theta = 2cos^{-1}(q_4), \tag{2.11}$$

and the axis of rotation would be given by

$$\hat{\mathbf{a}} = \frac{(q_1, q_2, q_3)^\top}{\sin(\theta/2)}. \tag{2.12}$$

Then, it can be shown that the elements of the rotation submatrix $\mathbf{R}$ are related to the orientation parameters $\hat{\mathbf{a}}$ and θ by

$$\hat{\mathbf{R}} = \begin{bmatrix} a_x^2 + (1 - a_x^2)c_\theta & a_x a_y c'_\theta - a_z s_\theta & a_x a_z c'_\theta + a_y s_\theta \\ a_x a_y c'_\theta + a_z s_\theta & a_y^2 + (1 - a_y^2)c_\theta & a_y a_z c'_\theta - a_x s_\theta \\ a_x a_z c'_\theta - a_y s_\theta & a_y a_z c'_\theta + a_x s_\theta & a_z^2 + (1 - a_z^2)c_\theta \end{bmatrix} \,, \tag{2.13}$$

where $s_\theta = \sin\theta$, $c_\theta = \cos\theta$, and $c'_\theta = 1 - \cos\theta$.

Once the rotation matrix $\hat{\mathbf{R}}$ is computed, we can use again the matching set of points to compute the translation vector $\hat{\mathbf{t}}$

$$\hat{\mathbf{t}} = \frac{1}{N}\left(\sum_{i=1}^{N} \mathbf{p}_1^{(i)} - \mathbf{R}\sum_{i=1}^{N} \mathbf{p}_2^{(i)}\right). \tag{2.14}$$

2.2.2 SVD-based Solution

This solution decouples the translational and rotation parts of the pose estimation problem by noting that, at the least-squares solution to Eq. 2.4, both of the two 3D point clouds should have the same centroid [14].

Thus, the rotation matrix is computed first by reducing the original least-squares problem to finding the rotation that minimizes

$$\sum_{i=1}^{N} \left\| \bar{\mathbf{p}}_1^{(i)} - \left(\mathbf{R}\bar{\mathbf{p}}_2^{(i)}\right) \right\|^2, \tag{2.15}$$

where

$$\bar{\mathbf{p}}_1^{(i)} = \mathbf{p}_1^{(i)} - \mathbf{c}_1 \tag{2.16}$$

and

$$\bar{\mathbf{p}}_2^{(i)} = \mathbf{p}_2^{(i)} - \mathbf{c}_2, \tag{2.17}$$

express the i-th point on the two point clouds translated to their corresponding centroids, with $\mathbf{c}_1$ and $\mathbf{c}_2$ the centroids of the first and the second point cloud, respectively.

In order to minimize Eq. 2.15, it is defined the 3×3 matrix $\mathbf{M}$

$$\mathbf{M} = \sum_{i=0}^{N} \bar{\mathbf{p}}_1^{(i)} \ \bar{\mathbf{p}}_2^{(i)^\top}, \tag{2.18}$$

where its singular value decomposition is given by

$$\mathbf{M} = \mathbf{U}\,\mathbf{\Sigma}\,\mathbf{V}^\top. \tag{2.19}$$

With this, the rotation matrix that minimizes Eq. 2.15 is

$$\hat{\mathbf{R}} = \mathbf{U}\,\mathbf{V}^\top \tag{2.20}$$

as long as $|\mathbf{U}\,\mathbf{V}^\top| = +1$. Otherwise, if $|\mathbf{U}\,\mathbf{V}^\top| = -1$, the solution is a reflection.

Finally, having found the rotation $\hat{\mathbf{R}}$, the translation is computed by

$$\hat{\mathbf{t}} = \mathbf{c}_1 - \hat{\mathbf{R}}\mathbf{c}_2. \tag{2.21}$$

2.3 Error Propagation

In this section we present a method to model the uncertainty of the relative motion measurements computed with the visual odometry approach just described in this Chapter. This method propagates the noise from each matching feature, along the visual odometry process, to end up with a relative pose covariance estimate.

One way to do this is by Monte Carlo simulation, however, this process is time-consuming. Instead, we opt for a closed-form computation based on first order error propagation. That is, given a continuously differentiable function $\mathbf{y} = \mathbf{f}(\mathbf{x})$ and the covariance $\mathbf{\Sigma}_x$ of the input $\mathbf{x}$, we can obtain the covariance $\mathbf{\Sigma}_y$ of the output $\mathbf{y}$ by linearizing $\mathbf{f}(\mathbf{x})$ around the expected value $\mathbf{x_o}$ by a first-order Taylor series expansion. Thus, the first-order error propagation to covariance $\mathbf{\Sigma}_y$ is given by

$$\mathbf{\Sigma}_y = \nabla\mathbf{f}\,\mathbf{\Sigma}_x\,\nabla\mathbf{f}^\top,$$

where $\nabla\mathbf{f}$ is the Jacobian of $\mathbf{f}$.

However, sometimes we might not have access to an explicit expression for $\mathbf{y} = \mathbf{f}(\mathbf{x})$, as it will be shown to be our case. Fortunately, though, we still can compute an expression for the Jacobian of $\mathbf{f}(\mathbf{x})$ by the implicit function theorem, which we introduce next.

The implicit function theorem can be stated as follows [7]:

Let $S \subset \mathbb{R}^n \times \mathbb{R}^m$ be an open set and let $\mathbf{\Phi} : S \to \mathbb{R}^m$ be a differentiable function. Suppose that $(\mathbf{x_o}, \mathbf{y_o}) \in S$ that $\mathbf{\Phi}(\mathbf{x_o}, \mathbf{y_o}) = \mathbf{0}$, and that $\left|\frac{\partial \mathbf{\Phi}}{\partial \mathbf{y}}\right|_{(\mathbf{x_o}, \mathbf{y_o})} \neq \mathbf{0}$. Then there is an open neighborhood $X \subset \mathbb{R}^n$ of $\mathbf{x_o}$, a neighborhood $Y \subset \mathbb{R}^m$ of $\mathbf{y_o}$, and a unique differentiable function $\mathbf{f} : X \to Y$ such that

$$\mathbf{\Phi}(\mathbf{x}, \mathbf{f}(\mathbf{x})) = \mathbf{0}$$

for all $\mathbf{x} \in X$.

This theorem tells us that $\mathbf{y} = \mathbf{f}(\mathbf{x})$ is implicitly defined by $\mathbf{\Phi}(\mathbf{x}, \mathbf{y}) = \mathbf{0}$. Then, if we differentiate $\mathbf{\Phi}$ with respect to $\mathbf{x}$ we get

$$\frac{\partial \mathbf{\Phi}}{\partial \mathbf{x}} + \frac{\partial \mathbf{\Phi}}{\partial \mathbf{f}}\frac{\mathbf{df}}{\mathbf{dx}} = 0.$$

From this expression we can notice that, by knowing $\mathbf{\Phi}$, we can compute the derivative of the function $\mathbf{f}$ with respect to $\mathbf{x}$, even though we do not have an explicit expression for it, that is,

$$\frac{\mathbf{df}}{\mathbf{dx}} = -\left(\frac{\partial \mathbf{\Phi}}{\partial \mathbf{y}}\right)^{-1} \frac{\partial \mathbf{\Phi}}{\partial \mathbf{x}}. \tag{2.22}$$

Next, $\mathbf{\Phi}$ can be computed as follows. If $\mathbf{y} = \mathbf{y}^*$ is a value where a cost function $C(\mathbf{x}, \mathbf{y})$ has a minimum, $\mathbf{\Phi}$ can be computed by the fact that, at the minimum of this cost function, $\frac{\partial C(\mathbf{x}, \mathbf{y}^*)}{\partial \mathbf{y}} = 0$, then we choose $\mathbf{\Phi} = \frac{\partial C}{\partial \mathbf{y}}$. Thus, by the implicit function theorem, in a neighborhood of $\mathbf{y}^*$ the Jacobian of $\mathbf{f}$ is

$$\nabla \mathbf{f} = -\left(\frac{\partial^2 C}{\partial \mathbf{y}^2}\right)^{-1} \left(\frac{\partial^2 C}{\partial \mathbf{y} \partial \mathbf{x}}\right)^{\top} \tag{2.23}$$

This is the case when the function $\mathbf{f}$ is involved in a cost function with no constraints, otherwise, determining $\mathbf{\Phi}$ takes additional steps.

For the visual odometry process just described, the error propagation is performed in two steps. In the first, the covariance of each matching point is propagated through the least-squares stereo reconstruction process to get the covariance estimate of the corresponding 3D scene point. In the second step, the covariance of each 3D point of the two point clouds that are aligned are propagated through the pose estimation process to finally obtain the covariance of the relative pose measurement.

First order error propagation requires the derivatives of a function that converts matching points into 3D points in the first step, and 3D point clouds into a pose in the last step. Although we do not have access to an explicit function for each step, implicit functions are given by each of the involved minimization processes. Next we show how we compute the ensuing Jacobians.

2.3.1 Matching Point Error Propagation

We want the covariance $\mathbf{\Sigma}_p$ of the 3D scene point $\mathbf{p} = [x, y, z]^{\top}$ given the covariance $\mathbf{\Sigma}_m$ of the left image matching feature $\mathbf{m} = [u, v]^{\top}$ and the covariance $\mathbf{\Sigma}_{m'}$ of the right image matching feature $\mathbf{m}' = [u', v']^{\top}$. For instance, if we are using SIFT descriptors, the scale at which each feature was found can be used as an estimate for its covariance.

Next, to find $\mathbf{\Sigma}_p$ we need to obtain a first-order propagation of the covariance of the uncorrelated matching image feature, which is given by

$$\mathbf{\Sigma}_p = \nabla \mathbf{g} \begin{bmatrix} \mathbf{\Sigma}_m & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{2\times 2} & \mathbf{\Sigma}_{m'} \end{bmatrix} \nabla \mathbf{g}^{\top} \tag{2.24}$$

where $\nabla \mathbf{g}$ is the Jacobian of the explicit function $\mathbf{g}$ that maps a pair of matching image points $\mathbf{u} = [u, v, u', v']^\top$ into its corresponding 3D scene point $\mathbf{p}$, i.e. $\mathbf{p} = \mathbf{g}(\mathbf{u})$.

As in this step the 3D scene point is found by solving the overdetermined system of equations given by Eq. 2.3, so as to apply the implicit function theorem, we need to express this process as an optimization problem. Thus, finding the 3D scene point $\mathbf{p}$ can be seen as minimizing the squared L_2-norm of the residual of Eq. 2.3, that is,

$$C(\mathbf{u}, \mathbf{p}) = \|\mathbf{A}\,\mathbf{p} - \mathbf{b}\|^2. \tag{2.25}$$

Computing the gradient of 2.25 with respect to $\mathbf{p}$ and setting it to zero, we find the minimum at

$$\mathbf{p}^* = (\mathbf{A}^\top \mathbf{A})^{-1}\,\mathbf{A}^\top \mathbf{b}, \tag{2.26}$$

assuming $\mathbf{A}$ to be invertible.

Lastly, having defined Eq. 2.25, by the implicit function theorem, the Jacobian of $\mathbf{g}$ is given by

$$\nabla \mathbf{g} = -\left(\frac{\partial^2 C}{\partial \mathbf{p}^2}\right)^{-1} \left(\frac{\partial^2 C}{\partial \mathbf{p} \partial \mathbf{m}}\right)^\top. \tag{2.27}$$

2.3.2 Point Cloud Error Propagation

In this step we are looking for the covariance $\mathbf{\Sigma}_d$ of the relative pose constraint $\mathbf{d}$ expressing the relative motion of the camera, given the covariances of each of the 3D points on the two point clouds. Here, again, this covariance will be computed by a first-order propagation, and we will need to compute the Jacobian of a function $\mathbf{h}$ that maps the points $\mathbf{P} = \left\{\mathbf{p}_1^{(i)}, \mathbf{p}_2^{(i)}\right\}$ on the two point clouds into the relative pose $\mathbf{d}$ that indicates the relative motion between the frame of the two clouds, i.e. $\mathbf{d} = \mathbf{h}(\mathbf{P})$.

If we express the relative pose $\mathbf{d}$ using Euler angles to represent its orientation, Eq. 2.4 can be written as follows

$$C(\mathbf{P}, \mathbf{d}) = \sum_{i=0}^{N} \left\| \mathbf{p}_1^{(i)} - \left(\mathbf{rot}(\phi_d, \theta_d, \psi_d)\, \mathbf{p}_2^{(i)} + \begin{bmatrix} x_d \\ y_d \\ z_d \end{bmatrix} \right) \right\|^2, \tag{2.28}$$

where $\mathbf{d} = [x_d, y_d, z_d, \phi_d, \theta_d, \psi_d]^\top$, $\mathbf{rot}(\phi_d, \theta_d, \psi_d)$ is the rotation matrix defined by the Euler angles, and N is the point cloud size. The optimal value for $\mathbf{d}$ is computed with either one of the two approaches described in Sect. 2.2.

Thus, with the implicit function theorem the Jacobian of $\mathbf{d} = \mathbf{h}(\mathbf{P})$ is given by

$$\nabla \mathbf{h} = -\left(\frac{\partial^2 C}{\partial \mathbf{d}^2}\right)^{-1} \left(\frac{\partial^2 C}{\partial \mathbf{d} \partial \mathbf{P}}\right)^\top. \tag{2.29}$$

Finally, the covariance $\mathbf{\Sigma}_d$ of the relative pose constraint $\mathbf{d}$ will be given by,

$$\mathbf{\Sigma}_d = \nabla \mathbf{h}\, \mathbf{\Sigma}_P\, \nabla \mathbf{h}^\top, \tag{2.30}$$

where $\mathbf{\Sigma}_P$ is the covariance of the two clouds of points $\mathbf{P}$, that is,

$$\mathbf{\Sigma}_P = \text{diag}\left(\mathbf{\Sigma}_{p1}^{(1)}, ..., \mathbf{\Sigma}_{p1}^{(N)}, \mathbf{\Sigma}_{p2}^{(1)}, ..., \mathbf{\Sigma}_{p2}^{(N)}\right), \tag{2.31}$$

which is a block diagonal matrix, where $\mathbf{\Sigma}_{p1}^{(i)}$ and $\mathbf{\Sigma}_{p2}^{(i)}$ are the covariances of the i-th point of the first and second clouds, respectively.

An alternative procedure would be to rely on optimization approaches to obtain the uncertainty in the pose estimation, similarly to [15]. However, in this work we opted instead for the use of the implicit function theorem to propagate uncertainties as it yields closed-form expressions.

2.3.3 Error Propagation Tests

The following tests evaluate whether the covariance resulted from the error propagation is consistent with N Monte Carlo runs, using both synthetic and real data. To this end, we compute the normalized state estimation error squared or NEES [16] for each Monte Carlo run

$$\epsilon_i = [\mathbf{s}_i - \mu]^\top\ \mathbf{\Sigma}^{-1}\ [\mathbf{s}_i - \mu] \tag{2.32}$$

and take the average

$$\bar{\epsilon} = \frac{1}{N} \sum_{i=0}^{N} \epsilon_i, \tag{2.33}$$

where $\mathbf{s}_i$ is the result of a Monte Carlo run, $\mathbf{\Sigma}$ the covariance obtained with the error propagation and μ is the solution to either Eq. 2.3, for the matching point error propagation, or Eq. 2.4, for the point cloud error propagation.

If the Monte Carlo runs are consistent with the error propagation results, then $N\bar{\epsilon}$ will have a Chi-Squared density with Nn_x degrees of freedom or $\chi^2_{Nn_x}$, where n_x is the dimension of $\mathbf{s}_i$ and χ^2_n denotes a Chi-Squared distribution of n degrees of freedom. We validate this using a Chi-square test with a two sided 95% probability region, defined by the interval $[l_1, l_2]$. Thus, if

$$N\bar{\epsilon} \in [l_1, l_2] \tag{2.34}$$

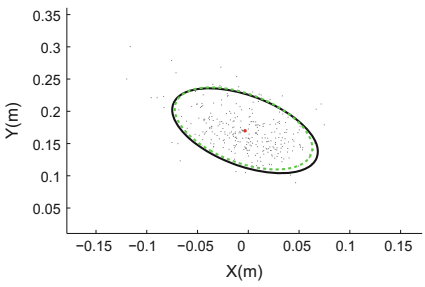
we confirm that the error propagation result is consistent with the Monte Carlo runs.

2.3.3.1 Synthetic Data

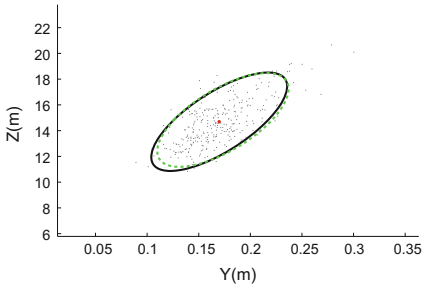
To test the matching point error propagation, we simulated a stereo camera system using the left camera as the reference frame, where the relative pose of the right camera with respect to the reference frame is given by a translation vector of $[0.385\,\text{m}, 0.007\,\text{m}, 0.00\,\text{m}]^\top$ and a rotation matrix expressed in Euler angles of $\mathbf{rot}(0.040\,\text{rad}, 0.016\,\text{rad}, 0.014\,\text{rad})$. Then, we simulated a ground truth 3D scene point and its corresponding imaged points in both cameras.

Next, we set the covariance $\boldsymbol{\Sigma}_m = \boldsymbol{\Sigma}_{m'} = \text{diag}(2\,\text{px}, 2\,\text{px})^2$ for both imaged points and apply the first-order error propagation (Eq. 2.24) to such covariances. Then, we perform a Monte Carlo simulation by generating a set of 300 pairs of random matching points around such image points and for each sample we obtain its corresponding 3D point with Eq. 2.26.

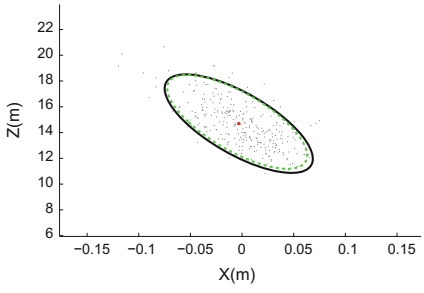
Figure 2.2 shows the simulated samples, the Monte Carlo covariance (black line), and the covariance computed with the error propagation (dashed green line). All hyperellipsoids represent iso-uncertainty curves plotted at a scale of 2 standard deviations.



(a) X-Y Plane covariance



(b) Y-Z Plane covariance



(c) X-Z Plane covariance

Fig. 2.2 Simulation of error propagation for the stereo reconstruction of a single image pair. The covariance obtained by Monte Carlo simulation is represented by the *black ellipse*, while the covariance computed with the first-order error propagation is plotted with the dashed *green ellipse*. All hyperellipsoids represent iso-uncertainty curves plotted at a scale of 2 standard deviations. The red point shows the mean of reconstructed 3D point

This test yielded $N\bar{\epsilon} = 860.1563$, lying within the interval [831.3, 970.4], which defines the two-sided 95% probability region for a χ^2_{900} variable, thus confirming the consistency of the error propagation.

To test the error propagation of the pose estimation process, we simulated two stereo systems with a known relative pose and placed 100 scene points uniformly distributed in the field of view of the four cameras and compute their corresponding imaged points, assigning to such points a covariance of $\mathbf{\Sigma}_m = \mathbf{\Sigma}_{m'} = \text{diag}(2\,\text{px}, 2\,\text{px})^2$. Then, we propagate their covariance along the whole visual odometry process.

Next, to perform a Monte Carlo simulation, with the covariance of each imaged points, we generate 1000 samples around each point, which yields 1000 point clouds. Then, we apply the least-squares best-fit pose transformation to each cloud.

Figure 2.3 shows the Monte Carlo covariance (black line), and the covariance computed with the implicit theorem function (dashed green line). All hyperellipsoids represent iso-uncertainty curves plotted at a scale of 2 standard deviations. From both Figs. 2.2 and 2.3, we can note that the covariances obtained with the first order error propagation are similar to covariances computed with Monte Carlo, with the advantage of the former being less expensive.

For this test we get $N\bar{\epsilon} = 6.07 \times 10^3$, which confirmed the consistency of the error propagation since it lies within the interval $\left[5.8 \times 10^3, 6.2 \times 10^3\right]$, defining the two-sided 95% probability region for a χ^2_{6000} variable.

2.3.3.2 Real Data

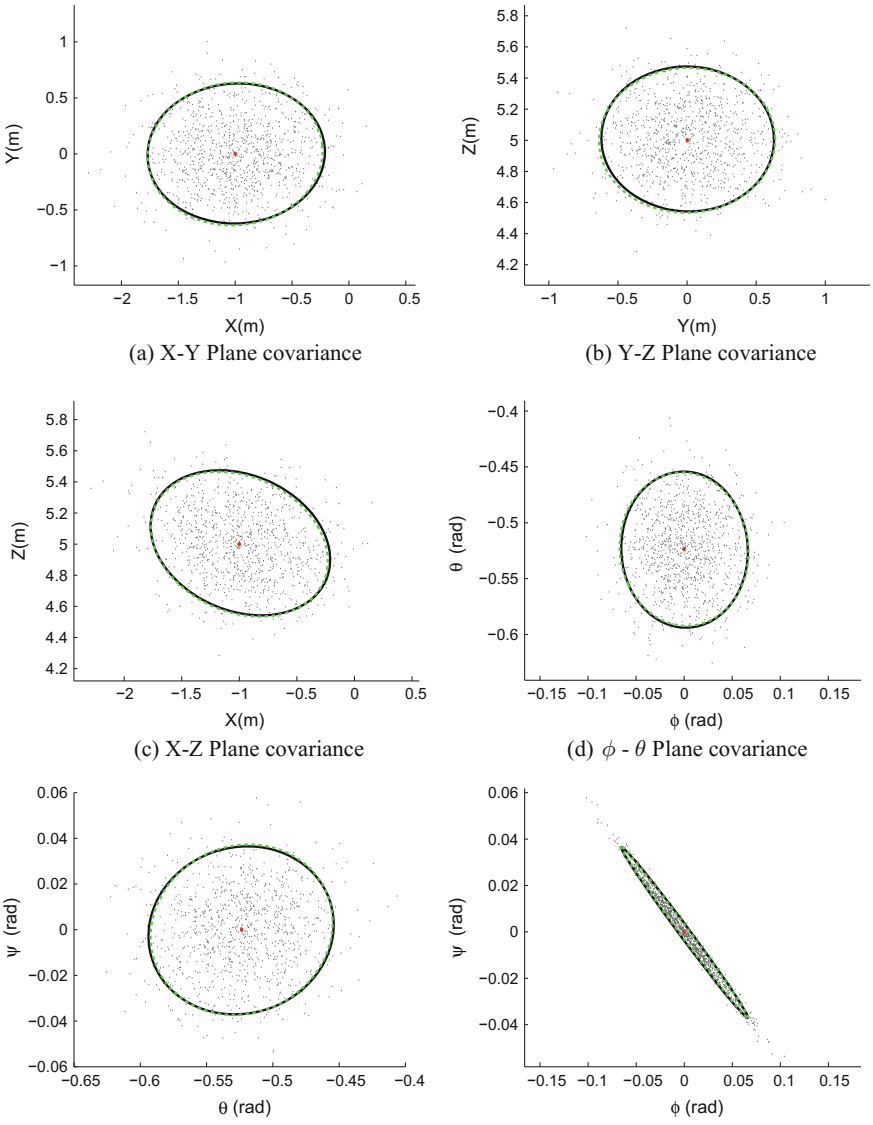
Next, we show a test using real data. To this end, we took a pair of stereo images out of a data set of stereo images in an outdoor environment. In the stereo camera system we used the left camera as the reference frame, as a result of the calibration process the relative pose of the right camera with respect to the left camera is given by a translation vector of $[0.385\,\text{m}, 0.007\,\text{m}, 0.00\,\text{m}]^\top$ and a rotation matrix expressed in Euler angles of $\mathbf{rot}(0.040\,\text{rad}, 0.016\,\text{rad}, 0.014\,\text{rad})$.

From these images we compute the relative motion between the stereo cameras and propagate the uncertainty of each matching feature through the whole visual odometry process. For our tests, the value of this covariance is approximated by the scale at which the SIFT where found. Figure 2.4 shows the ellipses representing iso-uncertainty curves plotted at a scale of 3 standard deviations. Next, we perform a Monte Carlo simulation taking 1000 runs.

Figure 2.5 shows the Monte Carlo covariance (black line), and the covariance computed with the implicit theorem function (dashed green line). All hyperellipsoids represent iso-uncertainty curves plotted at a scale of 2 standard deviations.

For this test we get $N\bar{\epsilon} = 6.14 \times 10^3$, lying in the aforementioned interval that defined the two-sided 95% probability region for a χ^2_{6000} variable. Thus confirming the consistency of the error propagation results with the Monte Carlo runs.

The above experiments showed the consistency of using a closed form expression to compute the covariance of the relative motion measurements obtained with stereo



(a) X-Y Plane covariance

(b) Y-Z Plane covariance

(c) X-Z Plane covariance

(d) ϕ - θ Plane covariance

(e) θ - ψ Plane covariance

(f) ϕ - ψ Plane covariance

Fig. 2.3 Simulation of the error propagation of the pose estimation from the two point clouds. The covariance obtained by Monte Carlo simulation is respresent by the *black ellipse*, while the covariance computed with the implicit function theorem is plotted with the dashed *green ellipse*. All hyperellipsoids represent iso-uncertainty curves plotted at a scale of 2 standard deviations. The *red point* shows the mean of the estimated pose

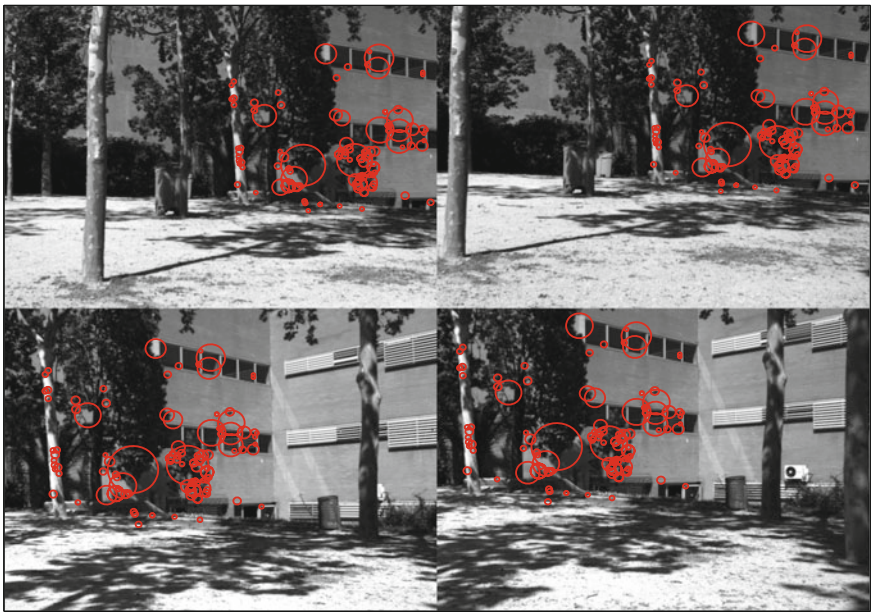
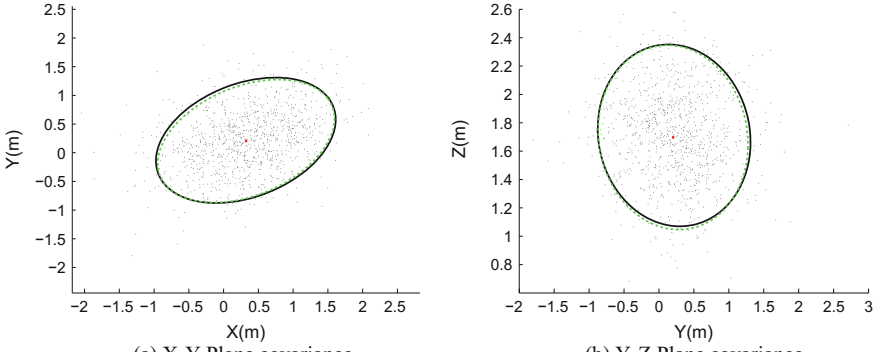


Fig. 2.4 Some of the SIFT correspondences in two consecutive stereo image pairs and their covariance. The ellipses represent iso-uncertainty curves plotted at a scale of 3 standard deviations

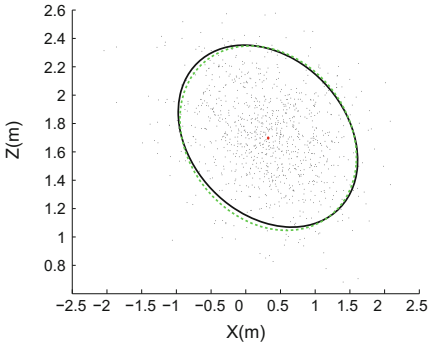
cameras. An alternative technique to model the error in motion measurements can be to use the method introduced by Matthies and Shafer in [15]. In their approach when the motion is purely translational, their technique exploits the linearity of such a case and obtains a direct solution, however, when the motion involves rotation the approach has to resort to an iterative solution and applies the Gauss-Newton method to refine iteratively the estimate.

Nevertheless, in some robotic applications it is desirable to use instead a closed-form expression to compute the covariance of the relative motion measurements, regardless of the type of motion. For instance, in map building applications modeling the uncertainty with a closed form expression is preferable. Although back-end SLAM approaches have evolved to the point of being very efficient, the bottle-neck of the map building process is still the front-end SLAM. Data association strategies usually require to compute relative motion measurements multiple times, thus a closed form approach would be preferable over iterative methods. For instance, for the case of laser range finders, a widely used Iterative Closest Point implementation in robotic applications introduced in [17] computes the covariance of the relative motion estimates with a similar closed form formulation to the one described in this Chapter.

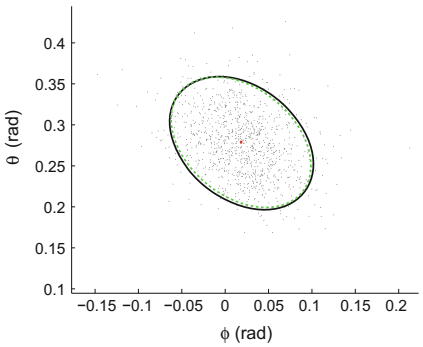


(a) X-Y Plane covariance

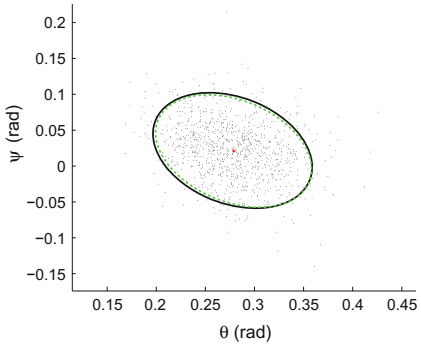
(b) Y-Z Plane covariance



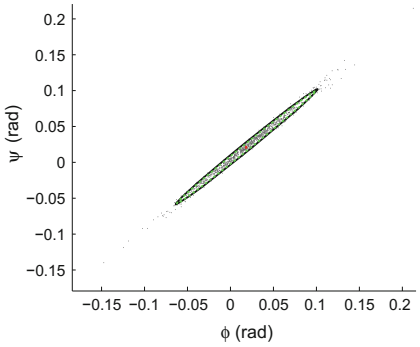
(c) X-Z Plane covariance



(d) ϕ - θ Plane covariance



(e) θ - ψ Plane covariance



(f) ϕ - ψ Plane covariance

Fig. 2.5 Error propagation of the relative pose estimation between two robot poses using stereo images. The covariance obtained by Monte Carlo simulation is respresent by the *black ellipse*, while the covariance computed with the implicit function theorem is plotted with the dashed *green ellipse*. All hyperellipsoids represent iso-uncertainty curves plotted at a scale of 2 standard deviations. The red point shows the mean of the estimated pose

2.4 Bibliographical Notes

Visual odometry is the problem of estimating the egomotion of the robot from one or multiple cameras attached to it. The term visual odometry was introduced in [18] but the problem of estimating the motion of a vehicle from visual imagery was previously addressed by Moravec in [19]. Visual odometry is a particular case of Structure From Motion or SFM [20], SFM seeks to recover the camera pose and three-dimensional structure from a set of images. It takes the 2D information and recover the original 3D information, inverting the effect of the projection process. Bundle adjustment (BA) is frequently used to improve upon SFM solutions [12].

Unlike SFM in visual odometry the 3D motion is estimated as a new frame arrives. The works on visual odometry have evolved in two branches, that is, in monocular and stereo visual odometry. Besides this, each work distinguishes in the way they solve each part of the problem, i.e., feature detection, feature matching, motion estimation, and local optimization. A comprehensive review on the evolution of these works appeared in [2, 3].

The work presented in this Chapter belongs to the stereo-based approaches to visual odometry. Most of these approaches track features along the frames obtained by corner detectors [19], such as Forstner [21, 22], Harris [23, 24] or Shi-Tomasi [25] or by selecting key points after performing dense stereo [26]. In these works 3D points are triangulated for every stereo pair, which are used to solve the motion estimation and incorporating RANSAC in the motion estimation for outlier removal. In the landmark work presented in [18], instead of tracking features among images the features are detected independently in all images. Additionally, instead of using corner detector, blob detectors has been also employed, such as SIFT in [27, 28], CENSURE in [29], SURF in [30], or BRIEF [31] descriptors in [32].

In visual odometry, errors in locating the image features lead to errors in the location of the 3D feature points after stereo reconstruction, which eventually cause errors in motion estimates. One of the first error modeling approaches for visual odometry was presented by Matthies and Shafer in [15]. In this work, the errors in the location of the image features are modeled as random variables following a 3D Gaussian distribution. Thus, the error in the location of 3D feature points are obtained by a first-order linear propagation of the covariances of image features. The error in motion estimates are computed by a maximum-likelihood estimation approach. Albeit for pure translation motion, this approach is an iterative solution, requiring an initial estimate. Although it has been applied successfully in spatial rovers as it is reported in [24], a closed-form expression for the motion uncertainty is preferred.

The implicit function theorem was initially exploited in [33, 34], where Chowdhury and Chellappa derive analytical expressions for the covariance of structure and motion estimates as a function of the covariance of the image correspondences. In [35], in order to estimate the uncertainty of the robot pose obtained via a correspondence-based method with stereo images, the authors employ the implicit function theorem to derive the pose uncertainty from a maximum likelihood formulation.

Besides visual odometry, the modeling of errors in pose estimation has been also addressed for wheel odometry [36–38] as well as for other sensor modalities. For range sensors, a model of the error of the pose estimation obtained with ICP is presented in [17] for laser range finders and in [39] for time-of-flight sensors. Both approaches are strongly related to the approach presented in this Chapter as they also model motion uncertainty by a first-order error propagation based on the implicit function theorem.

References

1. P.J. Besl, N.D. McKay, A method for registration of 3D shapes. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2), 239–256 (1992). Feb
2. D. Scaramuzza, F. Fraundorfer, Visual odometry: Part I—the first 30 years and fundamentals. Robot. Automat. Mag. **18**(4), 80–92 (2011). Dec
3. F. Fraundorfer, D. Scaramuzza, Visual odometry: Part II—matching, robustness, and applications. Robot. Automat. Mag. **19**(2), 78–90 (2012)
4. V. Ila, J. Andrade-Cetto, R. Valencia, A. Sanfeliu, Vision-based loop closing for delayed state robot mapping, in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots Systems* (San Diego, 2007), pp. 3892–3897
5. D.G. Lowe, Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004). Nov
6. M. Fischler, R. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Comm. ACM **24**, 381–385 (1981)
7. O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint* (The MIT Press, Cambridge, 1993)
8. C.G. Harris, M. Stephens, A combined corner edge detector, in *Proceedings of Alvey Vision Conference* (Manchester, 1988), pp. 189–192
9. J. Shi, C. Tomasi, Good features to track, in *Proceedings of 9th IEEE Conference on Computer Vision and Pattern Recognition* (Seattle, 1994), pp. 593–600
10. A.J. Davison, I.D. Reid, N.D. Molton, O. Stasse, MonoSLAM: Real-time single camera SLAM. IEEE Trans. Pattern Anal. Mach. Intell. **26**(6), 1052–1067 (2007)
11. K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**(10), 1615–1630 (2005)
12. R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision* (Cambridge University Press, Cambridge, 2000)
13. B.K.P. Horn, H.M. Hilden, S. Negahdaripour, Closed-form solution of absolute orientation using orthonormal matrices. J. Opt. Soc. Am. A **5**(7), 1127–1135 (1988)
14. K.S. Arun, T.S. Huang, S.D. Blostein, Least-squares fitting of two 3-D point sets. IEEE Trans. Pattern Anal. Mach. Intell. **9**(5), 698–700 (1987). Sep
15. L. Matthies, S. Shafer, Error modeling in stereo navigation. IEEE J. Robot. Autom. **3**(3), 239–248 (1987). Jun
16. Y. Bar-Shalom, X. Rong Li, T. Kirubarajan, *Estimation with Applications to Tracking and Navigation* (Wiley, New York, 2001)
17. A. Censi, An accurate closed-form estimate of ICP's covariance, in *Proceedings of the IEEE International Conference on Robotics and Automation* (Rome, 2007), pp. 3167–3172
18. D. Nister, O. Naroditsky, J. Bergen, Visual odometry, in *Proceedings of 18th IEEE Conference on Computer Vision and Pattern Recognition* (Washington, 2004), pp. 652–659
19. H.P. Moravec, Obstacle avoidance and navigation in the real world by a seeing robot rover. Ph.D. thesis, Stanford University, California, 1980

20. H.C. Longuet-Higgins, A computer program for reconstructing a scene from two projections. Nature **293**(11), 133–135 (1981)
21. C. Olson, L. Matthies, M. Schoppers, M. Maimone, Robust stereo ego-motion for long distance navigation, in *Proceedings of 14th IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2 (Hilton Head, SC, 2000), pp. 453–458
22. C. Olson, L. Matthies, M. Schoppers, M. Maimone, Rover navigation using stereo ego-motion. Robot. Auton. Syst. **43**(4), 215–229 (2003)
23. Y. Cheng, M. Maimone, L. Matthies, Visual odometry on the mars exploration rovers—a tool to ensure accurate driving and science imaging. Robot. Automat. Mag. **13**(2), 54–62 (2006)
24. M. Maimone, Y. Cheng, L. Matthies, Two years of visual odometry on the mars exploration rovers. J. Field Robot. **24**(3), 169–186 (2007)
25. A. Milella, R. Siegwart, Real-time scattering compensation for time-of-flight camera, in *International Conference on Computer Vision Systems* (2006), pp. 21–24
26. S. Lacroix, A. Mallet, R. Chatila, L. Gallo, Real-time scattering compensation for time-of-flight camera, in *Proceedings of International Symposium on Artificial Intelligence, Robotics and Automation in Space* (Noordwijk, 1999), pp. 433–440
27. S. Se, D. Lowe, J. Little, Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. Int. J. Robot. Res. **21**(8), 735–758 (2002). Aug
28. S. Se, D. Lowe, J. Little, Vision-based global localization and mapping for mobile robots. IEEE Trans. Rob. **21**(3), 364–375 (2005). Jun
29. K. Konolige, M. Agrawal, J. Solà, Large scale visual odometry for rough terrain, in *Proceedings of 13th International Symposium of Robotics Research* (Hiroshima, 2007)
30. L. Kneip, M. Chli, R. Siegwart, Robust real-time visual odometry with a single camera and an IMU, in *Proceedings of British Machine Vision Conference* (BMVA Press, Dundee, 2011), pp. 20.1–20.10
31. M. Calonder, V. Lepetit, C. Strecha, P. Fua, BRIEF: binary robust independent elementary features, in *Proceedings of 11th European Conference on Computer Vision* (2010), pp. 778–792
32. T. Kazik, L. Kneip, J. Nikolic, M. Pollefeys, R. Siegwart, Real-time 6D stereo visual odometry with non-overlapping fields of view, in *IEEE Conference on Computer Vision and Pattern Recognition* (Providence, 2012), pp. 1529–1536
33. A.R. Chowdhury, R. Chellappa, Statistical error propagation in 3D modeling from monocular video, in *IEEE CVPR Workshops* (Madison, 2003)
34. A.R. Chowdhury, R. Chellappa, Stochastic approximation and rate-distortion analysis for robust structure and motion estimation. Int. J. Comput. Vis. **55**(1), 27–53 (2003)
35. P. Zhang, J. Gu, E. Milios, Registration uncertainty for robot self-localization in 3D, in *Proceedings of 2nd Canadian Conference on Computer and Robot Vision* (Victoria, 2005), pp. 490–497
36. J. Borenstein, L. Feng, Measurement and correction of systematic odometry errors in mobile robots. IEEE Trans. Rob. **12**(6), 869–880 (1996). Dec
37. A. Martinelli, The odometry error of a mobile robot with a synchronous drive system. IEEE Trans. Rob. **18**(3), 399–405 (2002). Jun
38. J.M. Mirats, J.L. Gordillo, C.A. Borja, A closed-form expression for the uncertainty in odometry position estimate of an autonomous vehicle. IEEE Trans. Rob. **21**(5), 1017–1022 (2005). Oct
39. S. Foix, G. Alenyà, J. Andrade-Cetto, C. Torras, Object modeling using a ToF camera under an uncertainty reduction approach, in *Proceedings of IEEE International Conference on Robotics and Automation* (Anchorage, 2010), pp. 1306–1312

Springer

http://www.springer.com/978-3-319-60602-6

Mapping, Planning and Exploration with Pose SLAM
Valencia, R.; Andrade-Cetto, J.
2018, XII, 114 p. 40 illus., 38 illus. in color., Hardcover
ISBN: 978-3-319-60602-6