Optimal Computation of 3-D Similarity: Gauss-Newton vs. Gauss-Helmert

Kenichi KANATANI* and Hirotaka NIITSUMA
Department of Computer Science, Okayama University
Okayama 700-8530 Japan

(Received November 28, 2011)

Because 3-D data are acquired using 3-D sensing such as stereo vision and laser range finders, they have inhomogeneous and anisotropic noise. This paper studies optimal computation of the similarity (rotation, translation, and scale change) of such 3-D data. We first point out that the Gauss-Newton and the Gauss-Helmert methods, regarded as different techniques, have similar structures. We then combine them to define what we call the modified Gauss-Helmert method and do stereo vision simulation to show that it is superior to either of the two in convergence performance. Finally, we show an application to real GPS geodetic data and point out that the widely used homogeneous and isotropic noise model is insufficient and that GPS geodetic data are prone to numerical problems.

1. Introduction

The task of autonomous robots to reconstruct the 3-D structure of the scene using stereo vision and simultaneously compute its location in the map of the environment, called SLAM (Simultaneous Localization and Mapping), is one of the central themes of robotics studies today. One of the fundamental techniques for this is to compute the 3-D motion of the robot between two time instances. This information is obtained by tracking a particular 3-D object to compute its rotation, translation, and scale change. A similar task occurs in reconstructing the entire shape of a 3-D object by 3-D sensing, for which multiple sensors are necessary, because one sensor can reconstruct only the part that is visible to it. Hence, we need to map a partial shape obtained from one sensor to a partial shape obtained from another by computing an appropriate similarity between them. The same task arises for geodetic measurement of the earth surface from multiple satellite sensor data [1, 5, 7, 18].

Thus, 3-D similarity estimation is an important problem in many engineering applications. To this end, many researchers have focused on accurate rotation estimation since 1980s. This is because translation can be estimated from the displacement of the centroid of the object, and the scale change is easily perceived from its changing size, while rotation estimation is not so straightforward in the presence of noise. However, almost all rotation estimation algorithms proposed in the past [3, 4, 9, 10, 13, 24] have assumed homogeneous and isotropic noise. This is unrealistic for 3-D data acquired by 3-D sensing such as stereo vision and laser/ultrasonic range finders, because the accuracy is usually different between the depth direction and the direction orthogonal to it, resulting in an inhomogeneous and anisotropic noise distribution depending on the position, orientation, and type of the sensor.

It is [21] who first pointed out the inevitable inhomogeneity and anisotropy of the noise in 3-D data and presented a 3-D rotation estimation scheme that takes it into account. They used a technique called renormalization, which iteratively removes statistical bias of reweight least squares [14]. Recently, [19] presented a numerical scheme for computing an exact maximum likelihood (ML) solution. [20] extended it to 3-D similarity estimation. They used for the optimization computation the Levenberg-Marquardt (LM) method, the most widely used standard optimization technique in the field of computer vision [8]. The LM is basically the Gauss-Newton method, to which the gradient descent principle is combined to ensure convergence [23].

In geodetic science, on the other hand, the Gauss-Helmert method [2, 17, 18] is popular for similarity estimation; Helmert himself was a geodesist, and the similarity transformation is sometimes referred to as the “Helmert transformation”. The Gauss-Helmert method is also used in some computer vision ap-
applications [6, 22]. However, the Gauss-Newton and Gauss-Helmert methods have been regarded as different principles, and no comparative studies of them are found, partly because they have been mainly used in different domains: the former in robotics and computer vision, the latter in geodetic science. In this paper, we reformulate the Gauss-Helmert method in a form that makes the comparison easier and point out that the two have a very similar mathematical structures. Then, we combine them to define what we call the “modified Gauss-Helmert method” and do stereo vision simulation to show that it is superior to either of the two in the convergence performance. Finally, we show an application to real GPS geodetic data and point out that the widely used homogeneous and isotropic noise model is insufficient and that the numerical problem sometimes regarded as a shortcoming of Gauss-Helmert method is due to the inherent nature of GPS geodetic data.

2. Maximum likelihood estimation

Suppose we are given 3-D position measurements \( r_\alpha \) and \( r'_\alpha, \alpha = 1, \ldots, N \), before and after a similarity motion. We model the measurement uncertainty by independent Gaussian noise of mean 0 and covariance matrices \( c^2 V_0[r_\alpha] \) and \( c^2 V_0[r'_\alpha] \), where \( c \), which we call the noise level, describes the magnitude and \( V_0[r_\alpha] \) and \( V_0[r'_\alpha] \) which we call the normalized covariance matrices, describe the directional characteristics of the noise. If the noise is isotropic and homogeneous, we can let \( V_0[r_\alpha] = V_0[r'_\alpha] = I \) (the unit matrix) for all \( \alpha \), but in general \( V_0[r_\alpha] \) and \( V_0[r'_\alpha] \) are not diagonal and also different from position to position.

Let \( r_\alpha \) and \( r'_\alpha \) be the true positions of \( r_\alpha \) and \( r'_\alpha \), respectively, that undergo a similarity of rotation \( R \), translation \( t \), and scale change \( s \). Their optimal estimation in the sense of ML is to minimize the Mahalanobis distance, which we hereafter call “residual” for simplicity (the multiplier 1/2 is merely for convenience),

\[
J = \frac{1}{2} \sum_{\alpha=1}^{N} (r_\alpha - \bar{r}_\alpha, V_0[r_\alpha]^{-1}(r_\alpha - \bar{r}_\alpha))
+ \frac{1}{2} \sum_{\alpha=1}^{N} (r'_\alpha - \bar{r}'_\alpha, V_0[r'_\alpha]^{-1}(r'_\alpha - \bar{r}'_\alpha)),
\]

with respect to \( \bar{r}_\alpha \) and \( \bar{r}'_\alpha \) subject to

\[
\bar{r}'_\alpha = S \bar{r}_\alpha + t,
\]

for some rotation \( R \), translation \( t \), and scale change \( s \). Here, we combine the rotation \( R \) and scale change \( s \) into a “scaled rotation” \( S = sR \) and express it in terms of the quaternion^1 \( q = (q_0, q_1, q_2, q_3)^\top \) as follows:

\[
S = \begin{pmatrix}
q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_0 + q_2q_3) & 2(q_0q_2 - q_1q_3) \\
2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_1q_3 + q_2q_0) & 2(q_2q_0 - q_1q_3) \\
2(q_1q_0 - q_2q_3) & 2(q_1q_3 + q_2q_0) & q_0^2 - q_1^2 - q_2^2 + q_3^2 & 2(q_0q_2 + q_1q_3) \\
2(q_0q_1 - q_2q_3) & 2(q_0q_3 + q_2q_1) & 2(q_1q_3 - q_0q_2) & q_0^2 - q_1^2 - q_2^2 - q_3^2
\end{pmatrix}.
\]

This matrix represents a rotation if \( q \) is normalized to unit norm [11]. If \( q \) is not restricted to a unit vector, the square norm \( ||q||^2 \) represents the scale change \( s \). This quaternion representation of similarity is well known in computer vision and robotics, and it is also used in geodesic recently [2].

Introducing to (1) Lagrange multipliers \( \lambda_\alpha \) for the constraint (2), we let

\[
\bar{J} = \frac{1}{2} \sum_{\alpha=1}^{N} (r_\alpha - \bar{r}_\alpha, V_0[r_\alpha]^{-1}(r_\alpha - \bar{r}_\alpha))
+ \frac{1}{2} \sum_{\alpha=1}^{N} (r'_\alpha - \bar{r}'_\alpha, V_0[r'_\alpha]^{-1}(r'_\alpha - \bar{r}'_\alpha))
- \sum_{\alpha=1}^{N} (\lambda_\alpha, \bar{r}'_\alpha - S \bar{r}_\alpha - t).
\]

The ML estimators of \( \bar{r}_\alpha, \bar{r}'_\alpha, q \) and \( t \) are obtained by letting the derivatives of (4) with respect to them be 0 and solving the resulting equations.

3. Gauss-Newton method

We first formulate the Gauss-Newton method, the most fundamental optimization technique for robotics and computer vision. Differentiating (4) with respect to \( \bar{r}_\alpha \) and \( \bar{r}'_\alpha \), we obtain

\[
\nabla_{\bar{r}_\alpha} \bar{J} = -V_0[r_\alpha]^{-1}(r_\alpha - \bar{r}_\alpha) + S^\top \lambda_\alpha,
\]

\[
\nabla_{\bar{r}'_\alpha} \bar{J} = -V_0[r'_\alpha]^{-1}(r'_\alpha - \bar{r}'_\alpha) - \lambda_\alpha.
\]

Letting these be 0 and solving for \( \bar{r}_\alpha \) and \( \bar{r}'_\alpha \), we have

\[
\bar{r}_\alpha = r_\alpha - V_0[r_\alpha]S^\top \lambda_\alpha, \quad \bar{r}'_\alpha = r'_\alpha + V_0[r'_\alpha]\lambda_\alpha.
\]

Substituting these into (2), we can obtain the Lagrange multipliers \( \lambda_\alpha \) in the form

\[
\lambda_\alpha = -W_\alpha e_\alpha,
\]

where we define

\[
e_\alpha = r'_\alpha - S \bar{r}_\alpha - t, \quad W_\alpha = (SV_0[r_\alpha]S^\top + V_0[r'_\alpha])^{-1}.
\]

Substituting (7) into (6) and substituting the resulting \( \bar{r}_\alpha \) and \( \bar{r}'_\alpha \) into (1), we can express the residual \( J \) in the following form:
we obtain the 3-metric part counts in a quadratic form. If we define the second term on the right because only the sym-

The symmetrization operator \( q \) where

where we define both sides, we obtain and differentiating

Differentiating (3) with respect to \( q_i \), \( i = 0, 1, 2, 3 \), we obtain

where we define

Letting

and differentiating \( V_\alpha W_\alpha = I \) with respect to \( q_i \) on both sides, we obtain

from which \( \partial W_\alpha / \partial q_i \) is expressed as

where \( S[\cdot] \) denotes symmetrization \( (S[A] = (A + A^\top)/2) \). Thus, the derivative of (9) with respect to \( q_i \) is

we can write (15) in the form

where the second term on the right means the 4-D vector with that term as the \( i \)th component, \( i = 0, 1, 2, 3 \). Differentiation of (9) with respect to \( t \) yields

Differerentiating (15) with respect to \( q_j \) and doing the Gauss-Newton approximation, i.e., omitting the terms containing \( e_\alpha \), we obtain the second derivative

From (18), the second derivative with respect to \( t \) is

Differerentiating (18) with respect to \( q_i \) and doing the Gauss-Newton approximation, we obtain the following mixed second derivative:

Using the matrix \( U_\alpha \) in (16), we can thus express the Hessian of the residual \( J \) in the following form (Gauss-Newton approximation):

Thus, we obtain the following procedure for the Gauss-Newton method.

1. Provide an initial guess of \( q \) and \( t \), and let \( J_0 = \infty \) (a sufficiently large number).
2. Compute the scaled rotation \( S \) in (3) for \( q \).
3. Compute the vectors \( e_\alpha \) and the matrices \( W_\alpha \) in (8), and evaluate the residual \( J \) in (9).
4. If \( J \approx J_0 \), return \( q \) and \( t \) and stop. Else, let \( J_0 \leftarrow J \).
5. Compute the matrices \( Q_i \) in (11) and the matrices \( U_\alpha \) in (16).
6. Solve the following 7-D linear equation:
\[
\begin{pmatrix}
\sum_{a=1}^{N} U_{a}^{\top} W_{a} U_{a} & \sum_{a=1}^{N} U_{a}^{\top} W_{a} \\
\sum_{a=1}^{N} U_{a}^{\top} W_{a} & \sum_{a=1}^{N} W_{a}
\end{pmatrix}
\begin{pmatrix}
\Delta q \\
\Delta t
\end{pmatrix}
= \begin{pmatrix}
\sum_{a=1}^{N} U_{a}^{\top} \lambda_{a} \\
\sum_{a=1}^{N} \lambda_{a}
\end{pmatrix}
\begin{pmatrix}
\sum_{a=1}^{N} W_{a} Q_{a} V_{0}^{\alpha} r_{a}^{0} S_{\lambda_{a}^{*}} \\
0
\end{pmatrix}.
\]
(23)

7. Update \( q \) and \( t \) as follows, and return to Step 2:
\[
q \leftarrow q + \Delta q, \quad t \leftarrow t + \Delta t.
\]
(24)

As described in textbooks of optimization, (23) and (24) mean approximating, via Taylor expansion, the residual \( J \) in (9) by a quadratic function around the current values of \( q \) and \( t \) and moving to the minimum of the approximated residual \( J \) until the iterations converge.

4. Gauss-Helmert method

We now formulate the Gauss-Helmert method popular in geodetic science [2, 6, 17, 18, 22]. Suppose we are given some approximations \( r_{a}^{0} \) and \( r_{a}^{0} \) of the true positions \( r_{a} \) and \( r_{a}^{0} \). Let \( q \) and \( t \) be the current estimates of the true solution \( q \) and \( t \). We write
\[
r_{a} = r_{a}^{0} + \Delta r_{a}, \quad r_{a}' = r_{a}'^{0} + \Delta r_{a}',
q = q + \Delta q, \quad t = t + \Delta t,
\]
(25)
and compute the correction terms \( \Delta r_{a}, \Delta r_{a}', \Delta q, \) and \( \Delta t \). Substituting (25) into (2), doing Taylor expansion, and omitting second and higher order terms in the correction terms, we obtain
\[
r_{a}'^{(0)} + \Delta r_{a}' = S(r_{a}'^{(0)} + \Delta r_{a}) + \sum_{i=0}^{3} \frac{\partial S}{\partial q_{i}}(q_{i} + t + \Delta t),
\]
(26)
If (25) and (26) are substituted into (4), we obtain
\[
\dot{J} = \frac{1}{2} \sum_{a=1}^{N} \left( r_{a} - r_{a}^{0} - \Delta r_{a} - V_{0}[r_{a}]^{-1}(r_{a} - r_{a}^{0} - \Delta r_{a}) \right) + \frac{1}{2} \sum_{a=1}^{N} \left( r_{a}' - r_{a}'^{(0)} - \Delta r_{a}' - V_{0}[r_{a}']^{-1}(r_{a}' - r_{a}'^{(0)} - \Delta r_{a}') \right)
\]
\[
- \sum_{a=1}^{N} \left( r_{a}'^{(0)} + \Delta r_{a}' - S(r_{a}'^{(0)} + \Delta r_{a}) \right)
- \sum_{i=0}^{3} \frac{\partial S}{\partial q_{i}}(q_{i} + \Delta q_{i}) + (q_{i} + t + \Delta t)).
\]
(27)
Differentiating this with respect to \( \Delta r_{a}, \Delta r_{a}', \Delta q, \Delta t \) and letting the results be 0, we have
\[
-V_{0}[r_{a}]^{-1}(r_{a} - r_{a}^{0} - \Delta r_{a}) + S_{\lambda_{a}^{*}} = 0,
-V_{0}[r_{a}']^{-1}(r_{a}' - r_{a}'^{(0)} - \Delta r_{a}') - \lambda_{a} = 0.
\]
7. Update \( r_\alpha^{(0)} \), \( q \) and \( t \) as follows, and return to Step 2:

\[
\begin{align*}
r_\alpha^{(0)} &= r_\alpha - V_0[r_\alpha]S^T \lambda_\alpha, \\
qu &= q + \Delta q, \\
t &= t + \Delta t. 
\end{align*}
\] (35)

5. Reduced Gauss-Helmert method

The above description gives an impression that the Gauss-Newton and Gauss-Helmert methods are very different disciplines. We now show that the Gauss-Helmert method can be expressed in a form very similar to the Gauss-Newton method. From (33), we can express \( \lambda_\alpha \) in the form

\[
\lambda_\alpha = W_\alpha (U_\alpha^{(0)} \Delta q + \Delta t - e_\alpha),
\] (36)

where \( W_\alpha \) is the matrix defined in (8). If (36) is substituted into the first and the second equalities in (28) to eliminate \( \lambda_\alpha \), we obtain linear equations only in \( \Delta q \) and \( \Delta t \). Hence, the procedure of the Gauss-Helmert method can also be written in the following form:

1. Provide an initial guess of \( q \) and \( t \), and let \( r_\alpha^{(0)} = r_\alpha \) and \( J_0 = \infty \) (a sufficiently large number).
2. Compute the scaled rotation \( S \) in (3) for \( q \).
3. Compute the vectors \( e_\alpha \) and the matrices \( W_\alpha \) in (8), and evaluate the residual \( J \) in (9).
4. If \( J \approx J_0 \), return \( q \) and \( t \) and stop. Else, let \( J_0 = J \).
5. Compute the matrices \( Q_i \) in (11) and the matrices \( U_\alpha^{(0)} \) in (31).
6. Solve the following 7-D linear equation:

\[
\left( \sum_{\alpha=1}^{N} U_\alpha^{(0)\top} W_\alpha U_\alpha^{(0)} \right) \left( \sum_{\alpha=1}^{N} U_\alpha^{(0)\top} W_\alpha e_\alpha \right) = \left( \sum_{\alpha=1}^{N} U_\alpha^{(0)\top} W_\alpha \right) \Delta q \quad \left( \sum_{\alpha=1}^{N} U_\alpha^{(0)\top} W_\alpha \right) \Delta t
\]

\[
\begin{align*}
\left( \sum_{\alpha=1}^{N} U_\alpha^{(0)\top} W_\alpha e_\alpha \right) \left( \sum_{\alpha=1}^{N} U_\alpha^{(0)\top} W_\alpha \right)
\end{align*}
\] (37)

7. Compute \( \lambda_\alpha \) by (36).
8. Update \( r_\alpha^{(0)} \), \( q \), and \( t \) as follows, and return to Step 2:

\[
\begin{align*}
r_\alpha^{(0)} &= r_\alpha - V_0[r_\alpha]S^T \lambda_\alpha, \\
qu &= q + \Delta q, \\
t &= t + \Delta t.
\end{align*}
\] (38)

This reduction of the algorithm reduces the memory usage in the computer; the effect is significant when the number \( N \) of the data is very large. Also, the similarity to the Gauss-Newton method becomes more apparent. Comparing (23) and (37), we see that the matrices \( U_\alpha \) of (16) are replaced by the matrices \( U_\alpha^{(0)} \) of (31). For the Gauss-Newton method, the right side of (23) is the gradient of the residual \( J \) with respect to \( q \) and \( t \), but the counterpart of the second term is missing on the right side of (37). However, when \( \Delta q = \Delta t = 0 \) at the time of the convergence of the Gauss-Helmert iterations, we see from (36) that \( \lambda_\alpha = -W_\alpha e_\alpha \). Hence, the first equation in (38) implies

\[
r_\alpha^{(0)} = r_\alpha + V_0[r_\alpha]S^T W_\alpha e_\alpha,
\] (39)

which coincides with (6) if (7) is substituted. Substitution of (39) into (31) shows

\[
U_\alpha^{(0)\top} W_\alpha e_\alpha = U_\alpha W_\alpha e_\alpha
\]

\[
+ 2 \left( e_\alpha, W_\alpha Q_0 V_0[r_\alpha]S^T W_\alpha e_\alpha \right).
\] (40)

Hence, we have

\[
U_\alpha^{(0)\top} W_\alpha e_\alpha = U_\alpha W_\alpha e_\alpha
\]

\[
+ 2 \left( e_\alpha, W_\alpha Q_0 V_0[r_\alpha]S^T W_\alpha e_\alpha \right).
\] (41)

If this is substituted, the right side of (37) coincides with that of (23). Since this is 0 when the iterations have converged, we see that the Gauss-Helmert method returns the same solution as the Gauss-Newton method. Since the matrix on the left side of (23) is the Gauss-Newton approximation of the Hessian of the residual \( J \), the matrix on the left side of (37) can also be viewed as an approximation of the Hessian of \( J \). If we call it the Gauss-Helmert approximation, the Gauss-Newton and the Gauss-Helmert approximations differ by \( O(e_\alpha) \).

6. Modified Gauss-Helmert method

It is easily seen that (6) of the Gauss-Newton method gives the maximum likelihood estimators of \( r_\alpha \) and \( r_\alpha' \), given the current estimates \( q \) and \( t \), while the first equation in (35) (or (38)) gives an iterative update using the Lagrange multipliers \( \lambda_\alpha \) of (36) expressed in the increments \( \Delta q \) and \( \Delta t \) of \( q \) and \( t \). Although the resulting \( r_\alpha^{(0)} \) converge to the first equation of (6), as we showed in the preceding section, we can expect higher accuracy if we directly use the first equation of (6) rather than the iterative update as in (35) or (38). The modified procedure is summarized as follows:

1. Provide an initial guess of \( q \) and \( t \), and let \( J_0 = \infty \) (a sufficiently large number).
2. Compute the scaled rotation \( S \) in (3) for \( q \).
3. Compute the vectors \( e_\alpha \) and the matrices \( W_\alpha \) in (8), and compute \( r_\alpha^{(0)} \) as follows:

\[
r_\alpha^{(0)} = r_\alpha + V_0[r_\alpha]S^T W_\alpha e_\alpha.
\] (42)
4. Evaluate the residual $J$ in (9). If $J \approx J_0$, return $q$ and $t$ and stop. Else, let $J_0 \leftarrow J$.

5. Compute the matrices $Q_i$ in (11) and the matrices $U_{\alpha}^{(0)}$ in (31).

6. Solve the following 7-D linear equation:

$$
(\sum_{\alpha=1}^{N} U_{\alpha}^{(0)T} W_{\alpha} U_{\alpha}^{(0)}) \Delta q = (\sum_{\alpha=1}^{N} U_{\alpha}^{(0)T} W_{\alpha} \Delta t) = \left(\sum_{\alpha=1}^{N} U_{\alpha}^{(0)T} W_{\alpha} e_{\alpha}\right).
$$

7. Update $r_{\alpha}^{(0)}$, $q$, and $t$ as follows, and return to Step 2:

$$q \leftarrow q + \Delta q, \quad t \leftarrow t + \Delta t \quad (44)$$

As shown in the preceding section, if the values of $r_{\alpha}^{(0)}$ defined in (42) are used, the right side of (43) coincide with that of (23). In other words, while the right side of (37) of the (reduced) Gauss-Helmert method converges to that of the Gauss-Newton method, the above modified Gauss-Helmert method uses, from the beginning, the same value as the Gauss-Newton method on the right side of (43). Consequently, the only difference between the Gauss-Newton and the Gauss-Helmert methods is whether the Gauss-Newton approximation or the Gauss-Helmert approximation is used on the left side.

7. Simulated Stereo Vision

7.1 Covariance evaluation

A curved grid surface is rotated around the world origin $O$ and translated after its scale is changed, as depicted in Fig. 1. The 3-D positions of the grid points are reconstructed by stereo vision before and after the similarity motion. The simulated stereo images are shown in Fig. 2. The image size is set to 500 x 800 pixels and the focal length to 600 pixels. The two cameras are positioned so that the disparity angles, or the parallax, of the world origin $O$ is $10^\circ$. We added independent Gaussian noise of mean 0 and standard deviation $\sigma$ pixels to the $x$ and $y$ coordinates of each of the grid points in these images and computed their 3-D positions $\hat{r}_\alpha$ and $\hat{r}'_{\alpha}$ by the method described in [15]. For optimal similarity estimation, we need to evaluate the normalized covariances $V_{0}[r_\alpha]$ and $V_{0}[r'_{\alpha}]$ of the reconstructed 3-D positions $\hat{r}_\alpha$ and $\hat{r}'_{\alpha}$. Following [16] and [19], we evaluated these as follows.

We fix an $XYZ$ world coordinate system and regard the reference camera position as placed at the coordinate origin $O$ with the optical axis aligned to the $Z$-axis. The image $xy$ coordinate system is defined in such a way that its origin $o$ is at the principal point (the intersection with the optical axis) and the $x$- and $y$-axis are parallel to the $X$- and $Y$-axis of the world coordinate system, respectively. Then, the camera is rotated around the world coordinate origin $O$ by $R$ (rotation matrix) and translated by $t$ from the reference position. We call $\{R, t\}$ the motion parameters of the camera. The camera imaging geometry is modeled by perspective projection with focal length $f$, projecting a 3-D point onto a 2-D point $(x, y)$ by the following relationship [8]:

$$x \simeq PX, \quad x \equiv \begin{pmatrix} x/f_0 \\ y/f_0 \\ 1 \end{pmatrix}, \quad P \equiv \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

The symbol $\simeq$ means equality up to a nonzero constant multiplier, and $f_0$ is a scale constant of approximately the image size for stabilizing finite length computation. The $3 \times 4$ projection matrix $P$ is given by

$$P = \begin{pmatrix} f/f_0 & 0 & 0 \\ 0 & f/f_0 & 0 \\ 0 & 0 & 1 \end{pmatrix} (R^T - R^T t), \quad (46)$$

where the aspect ratio is assumed to be 1 with no image skews, or so corrected by prior calibration.

We consider two cameras with motion parameters $\{R, t\}$ and $\{R', t'\}$ with focal lengths $f$ and $f'$, respectively. Let $P$ and $P'$ be the projection matrices of the respective cameras, and $x$ and $x'$ the images of a point in 3-D observed by the respective cameras. Image processing for correspondence detection entails uncertainty to some extent, and we model it by independent isotropic Gaussian noise of mean 0 and standard deviation $\sigma$ (pixels). Due to noise, the
detected points $x$ and $x'$ do not exactly satisfy the epipolar constraint (Appendix A), so we correct $x$ and $x'$, respectively, to $\hat{x}$ and $\hat{x}'$ that exactly satisfy the epipolar constraint in an optimal manner (Appendix B). From the corrected positions $\hat{x}$ and $\hat{x}'$, the corresponding 3-D position $\hat{r}$ is uniquely determined (Appendix A). Note that although the noise in $x_0$ and $x_0'$ is assumed to be independent, the noise in the corrected positions $\hat{x}_0$ and $\hat{x}_0'$ is no longer independent [14]. The normalized covariance matrices $V_0[\hat{x}]$ and $V_0[\hat{x}']$ and the normalized correlation matrices $V_0[\hat{x},\hat{x}']$ and $V_0[\hat{x}',\hat{x}]$ are given as follows [14, 16]:

\[
V_0[\hat{x}] = \frac{1}{f_0}(P_k - \frac{(P_kF\hat{x}')(P_kF\hat{x}')^T}{\|P_kF\hat{x}'\|^2 + \|P_kF\hat{x}'\|^2}),
\]
\[
V_0[\hat{x}'] = \frac{1}{f_0}(P_k - \frac{(P_kF\hat{x}')(P_kF\hat{x}')^T}{\|P_kF\hat{x}'\|^2 + \|P_kF\hat{x}'\|^2}),
\]
\[
V_0[\hat{x},\hat{x}'] = \frac{1}{f_0}(-\frac{(P_kF\hat{x}')(P_kF\hat{x}')^T}{\|P_kF\hat{x}'\|^2 + \|P_kF\hat{x}'\|^2})
\quad = V_0[\hat{x}',\hat{x}]^T.
\]

Here, $F$ is the fundamental matrix between the two images (see Appendix A), and we define $P_k \equiv \text{diag}(1,1,0)$. Since the vector $\hat{X}$ reconstructed from $\hat{x}$ and $\hat{x}'$ satisfies the projection relationship in (45), vectors $\hat{x}$ and $P\hat{X}$ are parallel, and so are $\hat{x}'$ and $P'\hat{X}$. Thus, we have

\[
\hat{x} \times P\hat{X} = 0, \quad \hat{x}' \times P'\hat{X} = 0
\]  

(48)

It follows that if the noise in $\hat{x}$ and $\hat{x}'$ is $\Delta \hat{x}$ and $\Delta \hat{x}'$, respectively, the noise $\Delta \hat{X}$ in $\hat{X}$ satisfies to a first approximation

\[
\Delta \hat{x} \times P\hat{X} + \hat{x} \times P\Delta \hat{X} = 0,
\]
\[
\Delta \hat{x}' \times P'\hat{X} + \hat{x}' \times P'\Delta \hat{X} = 0.
\]

These are combined into one equation in the form

\[
\begin{pmatrix}
\hat{x} \times \hat{P} \\
\hat{x}' \times \hat{P}'
\end{pmatrix}
= \begin{pmatrix}
(P\hat{X}) \times I & O \\
O & (P'\hat{X}) \times I
\end{pmatrix}
\begin{pmatrix}
\Delta \hat{x} \\
\Delta \hat{x}'
\end{pmatrix},
\]

(50)

where $\Delta \hat{r}$ is the 3-D vector of the first three components of $\Delta \hat{X}$ and $\hat{P}$ and $\hat{P}'$ are the left $3 \times 3$ submatrices of the $3 \times 4$ projection matrices $P$ and $P'$, respectively. Here, we define the product $a \times A$ of a 3-D vector $a$ and a $3 \times 3$ matrix $A$ to be the $3 \times 3$ matrix whose columns are the vector products of $a$ and the respective columns of $A$ [14]. Multiplying both sides by the transpose of the left side from left, we obtain

\[
\begin{pmatrix}
\hat{x} \times \hat{P} \\
\hat{x}' \times \hat{P}'
\end{pmatrix}\Delta \hat{r} = \begin{pmatrix}
(P\hat{X}) \times I & O \\
O & (P'\hat{X}) \times I
\end{pmatrix}
\begin{pmatrix}
\Delta \hat{x} \\
\Delta \hat{x}'
\end{pmatrix}.
\]

The following identities hold [14]:

\[
\begin{align*}
(\hat{x} \times \hat{P})^T(\hat{x} \times \hat{P}) &= \hat{P}^T(\hat{x} \times I)^T(\hat{x} \times I)\hat{P} \\
&= \|\hat{x}\|^2\hat{P}^TP_N[\hat{x}]\hat{P}, \\
(\hat{x}' \times \hat{P}')^T(\hat{x}' \times \hat{P}') &= \hat{P}'^T(\hat{x}' \times I)^T(\hat{x}' \times I)\hat{P}' \\
&= \|\hat{x}'\|^2\hat{P}'^TP_N[\hat{x}'][\hat{P}'.
\end{align*}
\]

(52)

Here, we define

\[
\begin{align*}
P_N[\hat{x}] &\equiv I - N[\hat{x}]N[\hat{x}]^T, \\
P_N[\hat{x}'] &\equiv I - N[\hat{x}'][N[\hat{x}']^T,
\end{align*}
\]

(53)

where $N[\cdot]$ denotes normalization to unit norm ($N[a] = a/\|a\|$). Similarly, we have

\[
\begin{align*}
(\hat{x} \times \hat{P})^T((P\hat{X}) \times I) &= \hat{P}^T((\hat{x}, P\hat{X})I - (P\hat{X})\hat{x}^T), \\
(\hat{x}' \times \hat{P}')^T((P'\hat{X}) \times I) &= \hat{P}'^T((\hat{x}', P'\hat{X})I - (P'\hat{X})\hat{x}'^T).
\end{align*}
\]

(54)

Using these, we can rewrite (51) in the following form:

\[
A\Delta \hat{r} = \begin{pmatrix}
\Delta \hat{x} \\
\Delta \hat{x}'
\end{pmatrix},
\]

\[
A \equiv \|\hat{x}\|^2\hat{P}^TP_N[\hat{x}]\hat{P} + \|\hat{x}'\|^2\hat{P}'^TP_N[\hat{x}'][\hat{P}',
\]

\[
B = \begin{pmatrix}
\hat{P}^T((\hat{x}, P\hat{X})I - (P\hat{X})\hat{x}^T) \\
\hat{P}'^T((\hat{x}', P'\hat{X})I - (P'\hat{X})\hat{x}'^T)
\end{pmatrix}.
\]

(55)

Hence, we obtain

\[
\Delta \hat{r}\Delta \hat{r}^T = A^{-1}B \begin{pmatrix}
\Delta \hat{x}\Delta \hat{x}^T \\
\Delta \hat{x}'\Delta \hat{x}'^T
\end{pmatrix} B^T(A^{-1})^T.
\]

(56)

Taking expectation on both sides, we obtain the normalized covariance matrix $V_0[\hat{r}]$ of the reconstructed position $\hat{r}$ in the following form:

\[
V_0[\hat{r}] = A^{-1}B \begin{pmatrix}
V_0[\hat{x}] & V_0[\hat{x}, \hat{x}'] \\
V_0[\hat{x}, \hat{x}'] & V_0[\hat{x}'][\hat{V}]
\end{pmatrix} B^T(A^{-1})^T.
\]

(57)

Evaluating the normalized covariance matrix $V_0[\hat{r}_0]$ in (57), we find that the uncertainty distribution has an ellipsoidal shape elongated in the depth direction, as illustrated in Fig. 1. The ratio of radii is, on average over all the points, 1.00 : 1.685 : 5.090 in the vertical, horizontal, and depth directions, respectively, meaning that the error in the depth direction is approximately five times as large as in the vertical direction. We actually measured this ratio by adding noise to the images many times and found that it is about 1.00 : 1.686 : 5.095, a very close value to the prediction by (57).
Table 1: Three examples of the decrease in the residual $J$ for the standard deviation $\sigma = 1.0, 2.0, 3.0$ (pixels) of the error added to the stereo images. Unchanged digits are underlined.

<table>
<thead>
<tr>
<th>$\sigma = 1.0$</th>
<th>Gauss-Newton</th>
<th>Gauss-Helmert</th>
<th>modified G-H</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>23368.98044646554</td>
<td>23368.98044646554</td>
<td>23368.98044646554</td>
</tr>
<tr>
<td>1</td>
<td>5923.5604558145</td>
<td>151.2986897231218</td>
<td>1285.065292480236</td>
</tr>
<tr>
<td>2</td>
<td>260.224019664706</td>
<td>138.7852882171576</td>
<td>157.4589569299990</td>
</tr>
<tr>
<td>3</td>
<td>138.639772443412</td>
<td>138.4925647492029</td>
<td>138.5000828752851</td>
</tr>
<tr>
<td>4</td>
<td>138.4925871308799</td>
<td>138.492504345684</td>
<td>138.492504345684</td>
</tr>
<tr>
<td>5</td>
<td>138.492495387721</td>
<td>138.492494558186</td>
<td>138.492494551441</td>
</tr>
<tr>
<td>6</td>
<td>138.49249515073</td>
<td>138.49249515073</td>
<td>138.49249515073</td>
</tr>
<tr>
<td>7</td>
<td>138.492495141190</td>
<td>138.492495141190</td>
<td>138.492495141190</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\sigma = 2.0$</th>
<th>Gauss-Newton</th>
<th>Gauss-Helmert</th>
<th>modified G-H</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>23705.92405252490</td>
<td>23705.92405252490</td>
<td>23705.92405252490</td>
</tr>
<tr>
<td>1</td>
<td>6631.05935257285</td>
<td>594.2288594040884</td>
<td>1561.554831323493</td>
</tr>
<tr>
<td>2</td>
<td>736.3892569773028</td>
<td>558.6320435827311</td>
<td>558.6320435827311</td>
</tr>
<tr>
<td>3</td>
<td>553.902729010044</td>
<td>553.093125338568</td>
<td>553.093125338568</td>
</tr>
<tr>
<td>4</td>
<td>553.107204698275</td>
<td>553.093125338568</td>
<td>553.093125338568</td>
</tr>
<tr>
<td>5</td>
<td>553.093143463093</td>
<td>553.093143463093</td>
<td>553.093143463093</td>
</tr>
<tr>
<td>6</td>
<td>553.09315497525</td>
<td>553.09315497525</td>
<td>553.09315497525</td>
</tr>
<tr>
<td>7</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
</tr>
<tr>
<td>8</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
</tr>
<tr>
<td>9</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
</tr>
<tr>
<td>10</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
</tr>
<tr>
<td>11</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
<td>553.093155346927</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\sigma = 3.0$</th>
<th>Gauss-Newton</th>
<th>Gauss-Helmert</th>
<th>modified G-H</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>24182.94641626991</td>
<td>24182.94641626991</td>
<td>24182.94641626991</td>
</tr>
<tr>
<td>1</td>
<td>1749.68353117275</td>
<td>1385.51532477767</td>
<td>2074.41942851112</td>
</tr>
<tr>
<td>2</td>
<td>1602.79279463083</td>
<td>1264.397843471071</td>
<td>1237.58307763944</td>
</tr>
<tr>
<td>3</td>
<td>1243.65946344691</td>
<td>1243.44020868650</td>
<td>1237.28888227893</td>
</tr>
<tr>
<td>4</td>
<td>1237.72524024192</td>
<td>1237.32373689136</td>
<td>1237.288729761082</td>
</tr>
<tr>
<td>5</td>
<td>1237.32754736445</td>
<td>1237.280379736924</td>
<td>1237.288728527720</td>
</tr>
<tr>
<td>6</td>
<td>1237.29225473387</td>
<td>1237.288833051470</td>
<td>1237.288728517618</td>
</tr>
<tr>
<td>7</td>
<td>1237.28050047267</td>
<td>1237.288731797017</td>
<td>1237.288728517540</td>
</tr>
<tr>
<td>8</td>
<td>1237.28875863381</td>
<td>1237.288729488894</td>
<td>1237.288728517547</td>
</tr>
<tr>
<td>9</td>
<td>1237.288731196586</td>
<td>1237.288828533946</td>
<td>1237.288728517535</td>
</tr>
<tr>
<td>10</td>
<td>1237.288728762133</td>
<td>1237.288728519423</td>
<td>1237.288728517535</td>
</tr>
<tr>
<td>11</td>
<td>1237.288728539869</td>
<td>1237.288728519423</td>
<td>1237.288728517535</td>
</tr>
<tr>
<td>12</td>
<td>1237.288728519574</td>
<td>1237.288728519574</td>
<td>1237.288728517535</td>
</tr>
<tr>
<td>13</td>
<td>1237.288728517725</td>
<td>1237.288728517725</td>
<td>1237.288728517535</td>
</tr>
<tr>
<td>14</td>
<td>1237.288728517554</td>
<td>1237.288728517554</td>
<td>1237.288728517535</td>
</tr>
<tr>
<td>15</td>
<td>1237.288728517547</td>
<td>1237.288728517547</td>
<td>1237.288728517535</td>
</tr>
</tbody>
</table>
7.2 Results

Table 1 shows how the residual $J$ of each method decreases with iterations. The standard deviation of the error added to the stereo images is $\sigma = 1.0, 2.0, 3.0$ (pixels), respectively. We started from the identity ($R = I$, $t = 0$, $s = 1$) and imposed no threshold for convergence: we stopped if $J$ stops decreasing. In each step, unchanged digits are underlined. What is marked is that the initial decrease in $J$ of the Gauss-Helmert method is conspicuous; it drops sharply and almost abruptly. In contrast, Gauss-Newton method reduces almost abruptly. In contrast, Gauss-Newton method converges even faster. For the 1997 data, their covariance matrices of these values are estimated using statistical regression analysis. For the 1998 data, their covariates in stable areas. The covariance matrices of these values are estimated using statistical regression analysis. For the 1997 data, their covariates in stable areas. The covariance matrices of these values are estimated using statistical regression analysis. For the 1998 data, their covariates in stable areas. The covariance matrices of these values are estimated using statistical regression analysis. For the 1997 data, their covariates in stable areas. The covariance matrices of these values are estimated using statistical regression analysis.

Figure 3 shows for various $\sigma$ the average number of iterations over independent 1000 trials, each time with different noise. We can see that the Gauss-Helmert method converges faster than the Gauss-Newton method and in most cases the modified Gauss-Helmert method converges even faster. For comparison, Fig. 3b shows the result initialized using the widely used homogeneous and isotropic noise model which assumes $V_0[r_\alpha] = V_0[r_\alpha] = I$ for all $\alpha$. In this case, we compute the centroids $r_\alpha$ and $r_\alpha'$ of the data $\{r_\alpha\}$ and $\{r_\alpha'\}$, respectively, and the deviations $\tilde{r}_\alpha = r_\alpha - r_\alpha$ and $\tilde{r}_\alpha' = r_\alpha' - r_\alpha'$ from their respective centroids. As is well known, the scale change $s$ can be estimated by

$$s = \frac{\sum_{\alpha=1}^{N} \|\tilde{r}_\alpha\|^2}{\sum_{\alpha=1}^{N} \|\tilde{r}_\alpha'\|^2},$$

and the rotation $R$ is computed from $\{\tilde{r}_\alpha\}$ and $\{\tilde{r}_\alpha'\}$ by the method of singular value decomposition (SVD) (Appendix C). The translation $t$ is determined from $t = r' - sRr_\alpha$. As we see from Fig. 3b, the Gauss-Newton method converges faster than the Gauss-Helmert method for such an accurate initialization when the noise in the data is small. However, the Gauss-Helmert method becomes more efficient as the noise increases. Yet, the modified Gauss-Helmert method is always the most efficient for all the noise levels.

We next evaluated the accuracy of the computed rotation $\hat{R}$, the translation $\hat{t}$, and the scale change $\hat{s}$. Let $\hat{R}$, $\hat{t}$, and $\hat{s}$ be their true values, respectively. We computed the rotation angle $\delta\Omega$ (in degree) of the relative rotation $\hat{R}R^{-1}$, the translation error $\delta t = \hat{t} - t$ and the scale change error $\delta s = \hat{s} - s$. Repeating this 1000 times with $\sigma$ fixed, each time using different image noise, we evaluated the RMS errors

$$E_R = \sqrt{\frac{1}{1000} \sum_{a=1}^{1000} (\delta t^{(a)})^2},$$

$$E_t = \sqrt{\frac{1}{1000} \sum_{a=1}^{1000} \|\delta t^{(a)}\|^2},$$

$$E_s = \sqrt{\frac{1}{1000} \sum_{a=1}^{1000} (\delta s^{(a)})^2},$$

where the superscript $(a)$ denotes the value of the $a$th trial. Figure 4 plots these for various $\sigma$. Since, as far as accuracy is concerned, there are no discernible differences between the Gauss-Newton, the Gauss-Helmert, and the modified Gauss-Helmert methods, we compared their accuracy with the use of the homogeneous and isotropic noise model. It is clearly demonstrated that accurate estimation cannot be done unless the inhomogeneity and anisotropy of the 3-D sensing data.

8. Real Data Example

Turkey is a country with frequent earthquakes, and researchers monitor the land deformation using GPS data. Table 2 shows the $X$, $Y$, and $Z$ coordinates (in meters) of five positions selected from a landslide area near Istanbul in October 1997 and March 1998. The absolute positions are corrected in reference to control points in stable areas. The covariance matrices of these values are estimated using statistical regression analysis. For the 1997 data, their covariance matrices (in the order listed in the table) are

$$\begin{pmatrix}
34 & 10 & 17 \\
10 & 12 & 7 \\
17 & 7 & 33
\end{pmatrix},
\begin{pmatrix}
234 & 83 & 136 \\
83 & 97 & 58 \\
136 & 58 & 245
\end{pmatrix},
\begin{pmatrix}
24 & 8 & 12 \\
8 & 10 & 6 \\
12 & 6 & 25
\end{pmatrix},$$

multiplied by $10^{-8}$. For the 1998 data,

$$\begin{pmatrix}
51 & 18 & 23 \\
18 & 18 & 13 \\
23 & 13 & 30
\end{pmatrix},
\begin{pmatrix}
323 & 140 & 159 \\
140 & 148 & 100 \\
150 & 100 & 218
\end{pmatrix},
\begin{pmatrix}
41 & 14 & 19 \\
14 & 16 & 11 \\
19 & 11 & 28
\end{pmatrix},$$

multiplied by $10^{-8}$. For the 1998 data,

$$\begin{pmatrix}
141 & 47 & 70 \\
47 & 49 & 38 \\
70 & 38 & 96
\end{pmatrix},
\begin{pmatrix}
59 & 20 & 29 \\
20 & 24 & 16 \\
29 & 16 & 43
\end{pmatrix}.$$
underlined). We see that the values fluctuate in the digits below the eighth or ninth place. This phenomenon has been known in geodesic science and has sometimes been regarded as a shortcoming of the Gauss-Helmert method. However, this is common to the three methods. We see that although the data in Table 2 have many digits, the four or five digits are common to all the five observation points and that the differences between 1997 and 1998 occur only in the last two or three digits. As is well known in numerical analysis, such data are very prone to the magnification of the finite length computation error. We conclude that for all the methods the value of $J$ in Table 3a has practically converged at the second iteration and that the subsequently fluctuates in the digits below the eighth or ninth place indicate the accuracy limit due to finite length computation (double precision in our case). Accordingly, the similarity solutions also converge up to six or seven digits. Table 3b lists the meaningful digits with utmost attainable accuracy, to which the solution using the conventional homogeneous and isotropic noise model agree at most in the first digit. This illustrates that accurate estimation cannot be done without using the optimal computation shown here.

9. Conclusions

Because 3-D data are acquired using 3-D sensing such as stereo vision and laser range finders, they have inhomogeneous and anisotropic noise. In this paper, we studied optimal computation of the similarity (rotation, translation, and scale change) of such 3-D data. We compared the two well known methods that are suitable for this purpose: the Gauss-Newton method widely regarded as the standard optimization technique by computer vision and robotics engineers, and the the Gauss-Helmert method popular among geodetic scientists. Our observations are summarized as follows:

1. The Gauss-Helmert method has a very similar form to the Gauss-Newton method. This
becomes evident if the unknowns are reduced to the similarity variables alone by eliminating out the Lagrange multipliers. We can view the Gauss-Helmert iterations a variant of the Gauss-Newton iterations with a special approximation of the Hessian of the residual, which may be called the “Gauss-Helmert approximation”.

2. In the course of the iterations, the Gauss-Helmert method sharply drops the residual at first, but the subsequent convergence is not necessarily fast. In contrast, the Gauss-Newton iterations continue continuously and steadily reduce the residual. Overall, the Gauss-Newton method is more efficient when the initialization is accurate and the data noise is low, but the Gauss-Helmert method becomes more efficient when the initialization is poor or the noise level is high.

Then, we combined the merits of the two methods to define the “modified Gauss-Helmert method”. We observed that:

3. The initial drop of the residual is not so sharp as the Gauss-Helmert method, but the convergence is smooth as the Gauss-Newton method.

4. Irrespective of the accuracy of the initialization or the noise level of the data, the convergence is always faster than the Gauss-Newton or the Gauss-Helmert method.

We also described how to evaluate the covariance matrices of the inhomogeneous and anisotropic noise associated with stereo vision. Finally, we showed an application to real GPS geodetic data and found that:

5. The widely used homogeneous and isotropic noise model is insufficient for accurate estimation.

6. The limited accuracy, for which the Gauss-Helmert method has sometimes been to blame, is due to the inherent nature of GPS geodetic data. It is expected that our proposed modified Gauss-Helmert will become a standard tool in both computer vision and robotics engineering and geodetic science.

Acknowledgments. The authors thank Orhan Akyilmaz of Istanbul Institute of Technology, Turkey for providing the GPS data and doing helpful discussions. They also thank Takuto Honda of Okayama University and Hiroki Hara of SANYO Electric Co. Ltd. for helping our numerical experiments. This work was supported in part by the Ministry of Education, Culture, Sports, Science, and Technology, Japan, under a Grant in Aid for Scientific Research (C 21500172).

References


following epipolar constraint [8]:

\[
\begin{pmatrix}
\frac{x}{f_0} \\
\frac{y}{f_0}
\end{pmatrix}, \begin{pmatrix}
\frac{x'}{f_0} \\
\frac{y'}{f_0}
\end{pmatrix} = 0,
\]

Here, the matrix \( F = (F_{ij}) \), called the fundamental matrix, is defined by

\[
F_{11} = \begin{pmatrix}
P_{11} & P_{12} & P_{13} & P_{14} \\
P_{21} & P_{22} & P_{23} & P_{24} \\
P_{31} & P_{32} & P_{33} & P_{34} \\
P_{41} & P_{42} & P_{43} & P_{44}
\end{pmatrix},
F_{12} = \begin{pmatrix}
P_{11} & P_{12} & P_{13} & P_{14} \\
P_{21} & P_{22} & P_{23} & P_{24} \\
P_{31} & P_{32} & P_{33} & P_{34} \\
P_{41} & P_{42} & P_{43} & P_{44}
\end{pmatrix},
F_{13} = \begin{pmatrix}
P_{11} & P_{12} & P_{13} & P_{14} \\
P_{21} & P_{22} & P_{23} & P_{24} \\
P_{31} & P_{32} & P_{33} & P_{34} \\
P_{41} & P_{42} & P_{43} & P_{44}
\end{pmatrix},
F_{13} = \begin{pmatrix}
P_{11} & P_{12} & P_{13} & P_{14} \\
P_{21} & P_{22} & P_{23} & P_{24} \\
P_{31} & P_{32} & P_{33} & P_{34} \\
P_{41} & P_{42} & P_{43} & P_{44}
\end{pmatrix},
F_{31} = \begin{pmatrix}
P_{11} & P_{12} & P_{13} & P_{14} \\
P_{21} & P_{22} & P_{23} & P_{24} \\
P_{31} & P_{32} & P_{33} & P_{34} \\
P_{41} & P_{42} & P_{43} & P_{44}
\end{pmatrix},
F_{33} = \begin{pmatrix}
P_{11} & P_{12} & P_{13} & P_{14} \\
P_{21} & P_{22} & P_{23} & P_{24} \\
P_{31} & P_{32} & P_{33} & P_{34} \\
P_{41} & P_{42} & P_{43} & P_{44}
\end{pmatrix},
\]

where \( P_{ij} \) and \( P'_{ij} \) are the \((ij)\) elements of the projection matrices \( P \) and \( P' \) of the first and the second camera, respectively, as defined in (46). If we let \( r = (X, Y, Z)^T \) be the 3-D point we are looking at, we obtain from the perspective projection relationship in (45)

\[
x = f_0^3 X + P_{12}X + P_{13}X + P_{14}f_0
\]
\[
y = f_0^3 X + P_{22}X + P_{23}X + P_{24}f_0
\]
\[
x' = f_0^3 P'_{11}X + P'_{12}X + P'_{13}X + P'_{14}f_0,
\]
\[
y' = f_0^3 P'_{21}X + P'_{22}X + P'_{23}X + P'_{24}f_0
\]

Clearing the fraction, we obtain the following linear equations:

\[
\begin{pmatrix}
xP_{31} - f_0P_{11} \\
xP_{32} - f_0P_{12} \\
xP_{33} - f_0P_{13} \\
xP_{34} - f_0P_{14}
\end{pmatrix}, \begin{pmatrix}
yp_{31} - f_0P_{21} \\
yP_{32} - f_0P_{22} \\
yP_{33} - f_0P_{23} \\
yP_{34} - f_0P_{24}
\end{pmatrix} = \begin{pmatrix}
X \\
Y \\
Z
\end{pmatrix}
\]

(63)
B. Optimal Triangulation

Let \((x, y)\) and \((x', y')\) be a pair of corresponding points between stereo images. Since correspondence detection by an image processing operations inevitably entails uncertainty to some degree, they do not necessarily satisfy the epipolar constraint in (60). Geometrically, this corresponds to the fact that their lines of sight intersect, i.e., (60) is satisfied. By “optimally”, we mean that the correction is done in such a way that the reprojection error \((\hat{x} - x)^2 + (\hat{y} - y)^2 + (x' - x')^2 + (y' - y')^2\) is minimized. This correction procedure goes as follows [15]:

1. Let \(E_0 = \infty\) (a sufficiently large number), \(\hat{x} = x, \hat{y} = y, x' = x', y' = y'\), and \(\hat{x} = \hat{y} = 0\), and express the fundamental matrix \(F = (F_1)\) as the 9-D vector \(f = (F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33})^T\).

2. Compute the following 9×9 matrix \(V_0[\xi]\) and the 9-D vector \(\xi^T\):

\[
V_0[\xi] = \begin{pmatrix}
\hat{x}^2 + x'^2 & \hat{x}' y' & f_0 x' & \hat{x} y & 0 \\
\hat{x}' y' & \hat{x}'^2 + y'^2 & f_0 y' & 0 & \hat{x} y \\
\hat{x} y & f_0 y' & f_0^2 & 0 & 0 \\
x' y & 0 & 0 & y'^2 + x'^2 & x' y' \\
x y & 0 & \hat{x}' y' & 0 & \hat{x}'^2 + y'^2 \\
0 & \hat{x} y & f_0 x' & f_0 y' & 0 \\
0 & f_0 x' & 0 & 0 & f_0 y' \\
0 & f_0 y' & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0
\end{pmatrix}
\]

3. Update \(\hat{x}, \hat{y}, \hat{x}', \hat{y}', \hat{x}, \hat{y}, \hat{x}', \) and \(\hat{y}'\) as follows:

\[
\begin{pmatrix}
\hat{x} \\
\hat{y}
\end{pmatrix} = \frac{f_0}{f_0^2} \begin{pmatrix}
F_{11} & F_{12} & F_{13} \\
F_{21} & F_{22} & F_{23}
\end{pmatrix} \begin{pmatrix}
\hat{x}' \\
\hat{y}'
\end{pmatrix},
\]

\[
\begin{pmatrix}
\hat{x} \\
\hat{y}
\end{pmatrix} = \frac{f_0}{f_0^2} \begin{pmatrix}
F_{11} & F_{12} & F_{13} \\
F_{21} & F_{22} & F_{23}
\end{pmatrix} \begin{pmatrix}
\hat{x}' \\
\hat{y}'
\end{pmatrix},
\]

\[
\hat{x} = x - \hat{x}, \hat{y} = y - \hat{y}, \hat{x}' = x' - \hat{x}', \hat{y}' = y' - \hat{y}'.
\]

4. Compute the reprojection error \(E\) by

\[
E = \hat{x}^2 + \hat{y}^2 + \hat{x}'^2 + \hat{y}'^2.
\]

If \(E \approx E_0\), return \((\hat{x}, \hat{y})\) and \((\hat{x}', \hat{y}')\) and stop. Else, set \(E_0 = E\) and go back to Step 2.

C. Homogeneous isotropic noise model

Various methods are known for optimally computing the 3-D rotation for homogeneous and isotropic noise [3, 9, 10, 13, 24], but all are mathematically equivalent. The simplest is the following method in terms of the singular value decomposition (SVD) [12]:

1. Compute the following correlation matrix \(N\) between the 3-D positions \(r_\alpha\) and \(r'_\alpha\) before and after the rotations:

\[
N = \sum_{\alpha=1}^N r'_\alpha r_\alpha^T.
\]

2. Compute the SVD of \(N\) in the form

\[
N = U \text{diag}(\sigma_1, \sigma_2, \sigma_3) V^T,
\]

where \(U\) and \(V\) are orthogonal matrices, and \(\sigma_1 \geq \sigma_2 \geq \sigma_3 (\geq 0)\) are the singular values.

3. Return the following rotation matrix:

\[
R = U \text{diag}(1, 1, \det(UV^T)) V^T.
\]