Sparse Levenberg-Marquardt algorithm.


The Levenberg-Marquardt (LM) algorithm minimizes the following error function in an iterative procedure

\[ e_{res}^2 = \|X - \hat{X}\|^2_{\Sigma_X} = (X - \hat{X})^\top \Sigma_X^{-1} (X - \hat{X}) = \epsilon^\top \Sigma_X^{-1} \epsilon . \]

In the \( t \)-th iteration \( X - \hat{X}_t = \epsilon_t \), an \( N \)-dimensional vector in Cartesian coordinates. The way \( \epsilon_t \) is written here will have no minus sign in the minimizations. Represents the error between the measured original values and the current estimate of the measurements. This means simultaneous minimization of the parameters and of the measurements. The value \( M \) from now on means *all the unknown* and not only the parameters.

The first order Taylor series gives for the next iteration

\[ \epsilon_{t+1} = X - f(\hat{P}_{t+1}) = X - f(\hat{P}_t) - J_{f|P_t}(\hat{P}_{t+1} - \hat{P}_t) = \epsilon_t - J_{f|P_t} \delta \]

where \( \delta = (\hat{P}_{t+1} - \hat{P}_t) \), is the adjustment of all the estimates. The Jacobian \( J_{f|P_t} \) is an \( N \times M \) matrix. The number of measurements in Cartesian coordinated times the number of estimates: dimension of the parameters space plus the measurement in homogeneous coordinates. Therefore \( N < M \).

The variables to be determined are in the \( M \) dimensional vector \( \delta \). In each iteration we have to take into account the constraints of the parameters to have only the essential parameters. Also, if the last element of an estimated measurement is not one, we will divide all the coordinates with it to compute in Cartesian coordinates the new \( \epsilon_{t+1} \).

The normal equation, with a shorter writing of the Jacobian, is

\[ (J_t^\top \Sigma_X^{-1} J_t) \delta = J_t^\top \Sigma_X^{-1} \epsilon_t . \]

The LM algorithm uses two different ways to converge to a minimum which is hopefully close to the global optimum. An augmented normal equations decides if the next iteration should be a Gauss-Newton or a gradient descent procedure

\[ [J_t^\top \Sigma_X^{-1} J_t + \lambda_t \text{diag}(J_t^\top \Sigma_X^{-1} J_t)] \delta_t = J_t^\top \Sigma_X^{-1} \epsilon_t \]
where $\lambda$ is a scalar multiplier with the initial value typically $\lambda_0 = 10^{-3}$. The $M$ updates are the diagonal values of the $M \times M$ square matrix.

In the $(t + 1)$-th iteration:

- **If the error did decrease in the previous iteration**, $\lambda$ decreases by $\lambda_{t+1} = \lambda_t/10$. For small $\lambda$-s the LM does a Gauss-Newton iteration with the updated Jacobian $(J^\top \Sigma_X^{-1} J) \delta = J^\top \Sigma_X^{-1} \epsilon$. The matrix $J^\top \Sigma_X^{-1} J$ is always positive semi-definite and the error in the estimate of the objective function seems to be close to local linear convergence. If the increment in $\delta$ is further decreasing this error, the Gauss-Newton part continues.

- **If the error did increase in the previous iteration**, $\lambda$ increases by $\lambda_{t+1} = 10\lambda_t$. For large $\lambda$-s the LM does gradient descent $\lambda \delta = J^\top \Sigma_X^{-1} \epsilon$. The right side of the equation is proportional with the gradient of the objective function, $\epsilon^\top \Sigma_X^{-1} \epsilon$, and since $\epsilon = X - \hat{X}$, the sign will be negative. If the increment in $\delta$ decreases the error, the method goes back to Gauss-Newton. If not, the $\lambda$ is further increased.

If the nondiagonal terms are insignificant with respect to the diagonal ones, each of the estimates $P = \{p_j\}_{j=1}^M$ can be estimated separately. The process is repeated till $||\delta||$ becomes smaller then a threshold.

The LM algorithm needs an initial estimate of the parameters, which can be obtained by total least squares. From the initial estimate of the parameters the initial estimate of the measurements is obtained. The LM algorithm is sensitive to initialization. If the initial estimates is very far from the solution, LM may not be converging to the wanted result.

In computer vision the parameters and the measurements can be separated. For example, you want to recover 10,000 points in 3D from 100 views of 2D images. That is $3 \times 10,000$ unknowns in 3D positions. You also need the camera parameters, which have $12 \times 100$ unknown. (In fact is only 11 parameters and the first camera is given.) There are more than 30,000 unknowns and inversion of more than $30000 \times 30000$ matrix. However, if at each iteration the cameras are solved first and only after that we solve for the 3D measurements, we will have to do only about $1100 \times 1100$ matrix and 3D measurements are obtained by back substitution, transforming into an upper-triangular matrix.
The sparsity assumption separates the "parameters" from the "estimated measurements"

\[ P = \begin{pmatrix} a \\ b \end{pmatrix} \]

where, \( a \) is the vector of parameters and \( b \) is the vector of measurements. The Jacobian matrix has a block structure of \( A \) as \((N \times \text{parameters})\) plus \( B \) as \((N \times \text{measurements homogeneous coordinates})\)

\[ J = \frac{\partial \hat{X}}{\partial P} = \begin{bmatrix} A \\ B \end{bmatrix} \quad A = \begin{bmatrix} \frac{\partial \hat{X}}{\partial a} \end{bmatrix} \quad B = \begin{bmatrix} \frac{\partial \hat{X}}{\partial b} \end{bmatrix} . \]

We would like the next iteration of the estimates to be zero. The first order Taylor series becomes

\[ J_f|_{P_t}(P_{t+1} - P_t) = X - f(P_t)\]

\[ [A_t B_t] \begin{pmatrix} \delta_{a,t} \\ \delta_{b,t} \end{pmatrix} = \epsilon_t \]

where \( \epsilon_t \) is the error we want to reduce in the squared Mahalanobis distance.

We discard the iteration number, and the normal equation is

\[ \begin{bmatrix} A^\top \Sigma_X^{-1} A & A^\top \Sigma_X^{-1} B \\ B^\top \Sigma_X^{-1} A & B^\top \Sigma_X^{-1} B \end{bmatrix} \begin{pmatrix} \delta_a \\ \delta_b \end{pmatrix} = \begin{pmatrix} A^\top \Sigma_X^{-1} \epsilon \\ B^\top \Sigma_X^{-1} \epsilon \end{pmatrix} . \]

The \( M \times M \) augmentation is only along the diagonal of the big Jacobian matrix and therefore only the starred matrices change with \( \lambda \)

\[ \begin{bmatrix} U^* & W \\ W^\top & V^* \end{bmatrix} \begin{pmatrix} \delta_a \\ \delta_b \end{pmatrix} = \begin{pmatrix} \epsilon_A \\ \epsilon_B \end{pmatrix} . \]

Identification of the matrices is immediate. In the upper-left corner are the parameters, in the lower-right corner are the estimated measurements and in the upper-right (equal to the lower-left) are the cross-covariances.

We assume that \( V^* \simeq B^\top \Sigma_X^{-1} B \) are always full rank and therefore invertable. We multiply both sides with matrix

\[ \begin{bmatrix} \mathbf{I}_{\text{para x para}} \\ 0_{\text{meas.h.c. x para}} \\ \mathbf{I}_{\text{meas.h.c. x meas.h.c.}} \end{bmatrix} \]

resulting in

\[ \begin{bmatrix} U^* - WV^*(-1)W^\top \\ 0 \\ V^* \end{bmatrix} \begin{pmatrix} \delta_a \\ \delta_b \end{pmatrix} = \begin{pmatrix} \epsilon_A - WV^*(-1)\epsilon_B \\ \epsilon_B \end{pmatrix} . \]
At each iteration the parameters are estimated from the equation
\[
\delta_{a,(t+1)} = (U^*_t - W_t V_t^{*(1)} W_t^T) + (\epsilon_{A,t} - W_t V_t^{*(1)} \epsilon_{B,t})
\]
and the measurements by back substitution of one coordinate after the other from
\[
V_t^* \delta_{b,(t+1)} = \epsilon_{B,t} - W_t^T \delta_{a,t}.
\]
The next iteration starts from
\[
P_{(t+1)} = \begin{pmatrix}
a + \delta_{a,(t+1)} \\
b + \delta_{b,(t+1)}
\end{pmatrix}
\]
and function of the error value, the algorithm does one of the two possibilities till convergence.

Let’s illustrate the process with an example. The simultaneous estimation of multiple projective cameras and the 3D points from several views of 2D data, is called bundle adjustment. Assume we have three cameras \(j = 1, 2, 3\) and in each view there are four 2D points \(i = 1, 2, 3, 4\) from 3D. A 3D points can be written for each view, \(X_i = (x^T_{i1}, ..., x^T_{i3})^T\). The points in the \(j\)-th view have the same, independent noise. The unknown vector contains the estimated cameras and the estimated 3D points:
\[
P = (a^T_1, a^T_2, a^T_3, b^T_1, b^T_2, b^T_3, b^T_4)^T.
\]
Take the \(i\)-th 2D point on the \(j\)-th camera. The 2D point depends only on the \(j\)-th camera. The Jacobian in the camera parameters \(a = (a^T_1, ..., a^T_3)^T\) for \(i\)-th 2D point \(A_i = \left[ \frac{\partial \hat X_i}{\partial a} \right]\) is zero for any other camera \(k \neq j\) because then \(A_{ik} = \frac{\partial \hat x_{ik}}{\partial a_k}\). The matrix is a diagonal matrix \(A_i = \text{diag}(A_{i1}, A_{i2}, A_{i3})\) in the cameras and each block has \(i = 1, ... 4\).

The Jacobian of the \(i\)-th 3D point \(b_i\) is \(B_i = \left[ \frac{\partial \hat X_i}{\partial b_i} \right]\) which decomposed in 2D points \(B_i = (B^T_{i1}, ..., B^T_{i3})^T\) has an element \(B_{ij} = \frac{\partial \hat x_{ij}}{\partial b_i}\). In the \(j\)-th camera, the matrix \(B_{kj} = \frac{\partial \hat x_{kj}}{\partial b_k}\) is zero if the point is \(k \neq i\). The matrix is diagonal \(\text{diag}(B_{1j}, ..., B_{4j})\) and the diagonal matrix is repeated for each of the \(j = 1, 2, 3\) cameras.
A total of $12 \times 3 + 3 \times 4$ estimates have to be recovered in each iteration. (The first camera can be a $3 \times 3$ unit matrix and a 3-vector of zeros instead of translation.)

In general, $12 \times \text{camera} \ll 3 \times \text{measurements}$.

The covariance of the estimated quantities is

$$\Sigma_{\hat{P}} = (J^T_{f|\hat{P}} \Sigma_{X}^{-1} J_{f|\hat{P}})^+$$

where no variation in the parameters is allowed in directions perpendicular to the constraint surface. Since the final estimates are used, the augmentation is no longer needed. The matrix can be diagonalized is this way

$$J^T_{f|\hat{P}} \Sigma_{X}^{-1} J_{f|\hat{P}} = \begin{bmatrix} U & W \\ W^T & V \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{pr \times pr} & \mathbf{WV}^{-1} \\ 0_{mhc \times pr} & \mathbf{I}_{mhc \times mhc} \end{bmatrix} \begin{bmatrix} \mathbf{U} - \mathbf{WV}^{-1} \mathbf{W}^T & 0 \\ 0 & \mathbf{V} \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 \\ \mathbf{V}^{-T} \mathbf{W}^T & \mathbf{I} \end{bmatrix}.$$ 

Two matrices, $\mathbf{G}$ invertible and $\mathbf{H}$, have the identity

$$(\mathbf{G} \mathbf{H} \mathbf{G}^T)^+ = \mathbf{G}^{-T} \mathbf{H}^+ \mathbf{G}^{-1}$$

under very relaxed null-space conditions. Writing

$$X = (\mathbf{U} - \mathbf{WV}^{-1} \mathbf{W}^T)^+ \quad Y = \mathbf{WV}^{-1}$$
we obtain
\[ \Sigma_\hat{P} = \begin{bmatrix} X & -XY \\ -Y^TX & Y^TXY + V^{-1} \end{bmatrix} . \]

See the matrix inversion in block form in the second part of the planar homography material. All the covariances are based on the measured data and not on other computed covariances.

The covariance for the (essential) parameters is
\[ \Sigma_\hat{a} = (U - W V^{-1} W^T)^+ \]
the covariance of the estimated measurements is
\[ \Sigma_\hat{b} = Y^T \Sigma_\hat{a} Y + V^{-1} \]
and the cross-covariance between the parameters and estimated measurements is
\[ \Sigma_{ab} = -\Sigma_\hat{a} Y . \]

**Sparse LM for 2D homography estimation.**

The corresponding homogeneous image points in the two images are \( x_i \leftrightarrow x'_i \), connected through a \( 3 \times 3 \) nonsingular matrix \( H \). The measurements are \( \hat{X}_i = (\hat{x}_i^T, \hat{x}'_i^T)^T \). In total there are \( n \) measurements in a \( 4n \) long column vector. The noise covariance for each point is independent, isotropic and equal. Therefore, also \( \Sigma^{-1}_{\hat{x}_i} = \Sigma^{-1}_{\hat{x}'_i} = S_2 \) and \( \Sigma^{-1}_{\hat{X}} \) is \( n \) times the \( \text{diag}(S_2, S_2) \), a \( 4n \times 4n \) matrix.

The Jacobian matrix for homography is an \( n \) time \( 4 \times 9 \) matrix
\[ A_i = \frac{\partial \hat{X}_i}{\partial h} = \begin{bmatrix} 0_{2\times9} \\ \frac{\partial \hat{x}'_i}{\partial h} \\ \frac{\partial \hat{x}'_i}{\partial h} \end{bmatrix} \]
since \( \hat{x}_i \) does not depend of \( h \). Note that \( \hat{X}_i \) is in Cartesian coordinates, and the precise formulation of the derivate was given at the beginning of the lecture.

The Jacobian matrix for the first image homogeneous coordinates is an \( n \) time \( 4 \times 3 \) matrix
\[ B_i = \frac{\partial \hat{X}_i}{\partial x_i} = \begin{bmatrix} I_{2\times3} \\ \frac{\partial \hat{x}'_i}{\partial x_i} \\ \frac{\partial \hat{x}'_i}{\partial x_i} \end{bmatrix} \]
The estimated homogeneous measurements for a pair is

\[ \hat{X}_i = (\hat{x}_i^\top, \hat{x}_i'\,^\top) = (\hat{x}_i^\top, (\hat{H}\hat{x}_i)^\top)^\top \]

and it is enough to estimate the parameters (9) and the points in the first image (3n), because the points in the second image are then determined

\[ P = (h^\top, \hat{x}_1^\top, \ldots, \hat{x}_n^\top)^\top. \]

The eight essential parameters, the 2n Cartesian coordinated in the first image and from there the coordinates in the second image, are computed in each iteration.

We will compute the covariance of the homography transformation in two simple cases. The first case is \( H = sI \).

\[ A_i = \begin{bmatrix} 0_{2 \times 9} \\ J_i \end{bmatrix} \quad B_i = \begin{bmatrix} I_{2 \times 3} \\ sI_{2 \times 3} \end{bmatrix} = \begin{bmatrix} I_{2 \times 2} \\ sI_{2 \times 2} \end{bmatrix} \]

since the last column in \( B_i \) is always zero.

The matrix \( U \) from the Jacobian is a 9 \( \times \) 9 matrix because \( A_i \) simplifies to 2 \( \times \) 9

\[ U = \sum_{i=1}^{n} A_i^\top \text{diag}(S_2, S_2) A_i = \sum_{i=1}^{n} J_i^\top S_2 J_i. \]

The matrix \( V = \text{diag}(V_1, \ldots, V_n) \) is 2n \( \times \) 2n matrix because \( B_i \) is only 4 \( \times \) 2 and only the first image coordinated are sought

\[ V_i = B_i^\top \text{diag}(S_2, S_2) B_i = (s^2 + 1)S_2. \]

The matrix \( W = [W_1, \ldots, W_n] \) is 9 \( \times \) 2n matrix, each element being 9 \( \times \) 2

\[ W_i = A_i^\top \text{diag}(S_2, S_2) B_i = sJ_i^\top S_2. \]

We have to recover 9 + 2n estimates at each iteration. The \( H \) being very simple, the estimation in Cartesian coordinate directly.

The covariance of the homography is

\[
\Sigma_h = (U - WV^{-1}W^\top)^+ = \left( \sum_{i=1}^{n} J_i^\top S_2 J_i - \sum_{i=1}^{n} (sJ_i^\top S_2) \frac{1}{s^2 + 1} S_2^{-1} (sS_2 J_i) \right)^+ 
\]

7
where we took into account the symmetry of the noise covariance. Gives a $9 \times 9$ matrix of rank eight

$$\Sigma_h = (s^2 + 1) \left( \sum_{i=1}^{n} J_i^\top S_2 J_i \right)^+.$$ 

In the second case $H$ is an affine transformation

$$H = \begin{bmatrix} D & t \\ 0^\top & 1 \end{bmatrix}$$

and the noise covariance is a unit matrix $S_2 = I_2$. In this case the second image is also directly in Cartesian coordinates and $B_i$ can be expressed in Cartesian coordinates too

$$A_i = \begin{bmatrix} 0_{2\times 9} \\ J_i \end{bmatrix} \quad B_i = \begin{bmatrix} I_{2\times 2} \\ D \end{bmatrix}.$$ 

The matrix $U$ is

$$U = \sum_{i=1}^{n} A_i^\top \text{diag}(I_2, I_2) A_i = \sum_{i=1}^{n} J_i^\top J_i.$$ 

The matrix $V_i$ of the $i$-th measurements is

$$V_i = B_i^\top \text{diag}(I_2, I_2) B_i = I + D^\top D.$$ 

The matrix $W_i$, the cross-covariance of the parameters and the $i$-th measurement, is

$$W_i = A_i^\top \text{diag}(I_2, I_2) B_i = J_i^\top D.$$ 

The covariance of the homography is

$$\Sigma_h = \left( \sum_{i=1}^{n} J_i^\top J_i - \sum_{i=1}^{n} (J_i^\top D)(I + D^\top D)^{-1}(D^\top J_i) \right)^+$$

or

$$\Sigma_h = \left( \sum_{i=1}^{n} J_i^\top (I - D(I + D^\top D)^{-1}D^\top) J_i \right)^+.$$