Minimization by a numerical algorithm

- The projected coordinates of j-th point in i-th frame depend on
  - parameters of ith camera and
  - spatial coordinated of j-th point.
- Numerical optimization by Levenberg-Marquardt algorithm.
  - Jacobian matrix of the problem has to be determined.
  - Jacobian is very sparse.
- Thus, a sparse Levenberg-Marquardt algorithm should be applied.
  - It is called bundle Adjustment (BA) in the literature.

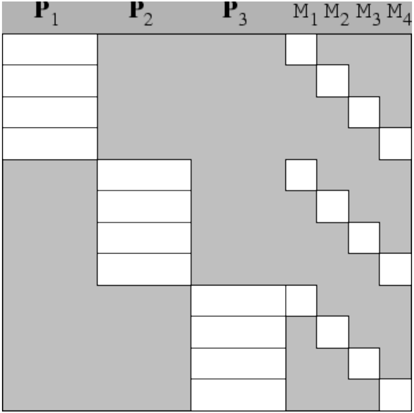
Levenberg-Marquardt for 3D Reconstruction

- LM-rule for parameter tuning:

$$\Delta \mathbf{p} = \left(\mathbf{J}^T \mathbf{J} + \lambda \mathbf{I}\right)^{-1} \mathbf{J}^T \epsilon_p$$

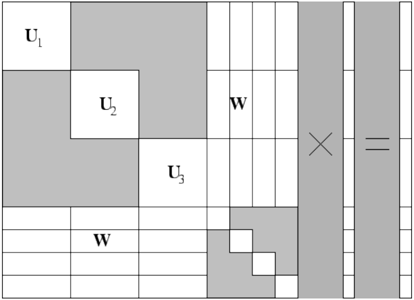
- Parameters to be tuned:
  - camera parameters
  - spatial coordinates
- E.g. for 20 perspective cameras and 1000 3D points: $20 \cdot 11 + 3 \cdot 1000 = 3220$ parameters have to be estimated
  - Dimension of $\mathbf{J}^T\mathbf{J}$ is 3220×3220.
  - Matrix invertion requires very high time demand.
  - Numerical stability of invertion is questionable

Jacobian matrix



Jacobian matrix

Jacobian matrix



Normal equation

Bundle adjustment: normal equation

- Normal equation can be written by block of matrices:

$$\begin{bmatrix} \mathbf{U} & \mathbf{X} \\ \mathbf{X}^T & \mathbf{V} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{m} \\ \Delta\mathbf{s} \end{bmatrix} = \begin{bmatrix} \epsilon_{\mathbf{m}} \\ \epsilon_{\mathbf{s}} \end{bmatrix}$$

- If normal equation is multiplied by $\begin{bmatrix} \mathbf{I} & -\mathbf{X}\mathbf{V}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$, from the left, normal equation is modified as follows:

$$\begin{bmatrix} \mathbf{U} - \mathbf{X}\mathbf{V}^{\mathbf{T}}\mathbf{X}^{\mathbf{T}} & \mathbf{0} \\ \mathbf{X}^{\mathbf{T}} & \mathbf{V} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{m} \\ \Delta\mathbf{s} \end{bmatrix} = \begin{bmatrix} \epsilon_{\mathbf{m}} - \mathbf{X}\mathbf{V}^{-1}\epsilon_{\mathbf{s}} \\ \epsilon_{\mathbf{s}} \end{bmatrix}$$

Bundle adjustment: solution for normal equation

- Solution:

$$\Delta \mathbf{m} = \left(\mathbf{U} - \mathbf{X}\mathbf{V}^{\mathbf{T}}\mathbf{X}^{\mathbf{T}}\right)^{-1} \left(\epsilon_{\mathbf{m}} - \mathbf{X}\mathbf{V}^{-\mathbf{1}}\epsilon_{\mathbf{s}}\right)$$

$$\Delta \mathbf{s} = \mathbf{V}^{-\mathbf{1}} \left(\epsilon_{\mathbf{s}} - \mathbf{X}^{\mathbf{T}}\Delta \mathbf{m}\right)$$

- Inversion required:
  - $\mathbf{V}$:
    - It contains small block matrices, they are inverted separately:
  - $\left(\mathbf{U} - \mathbf{X}\mathbf{V}^{-\mathbf{1}}\mathbf{X}^{\mathbf{T}}\right)^{-1}$
    - Its size is relatively small.
    - Is is also a special matrix, sub-blocks can be formed.