SFM and Bundle adjustment

1.) Structure from motion

- Problem statement.
- Affine SFM.
- Projective SFM.
- Understand the ambiguities.

2.) Bundle adjustment.

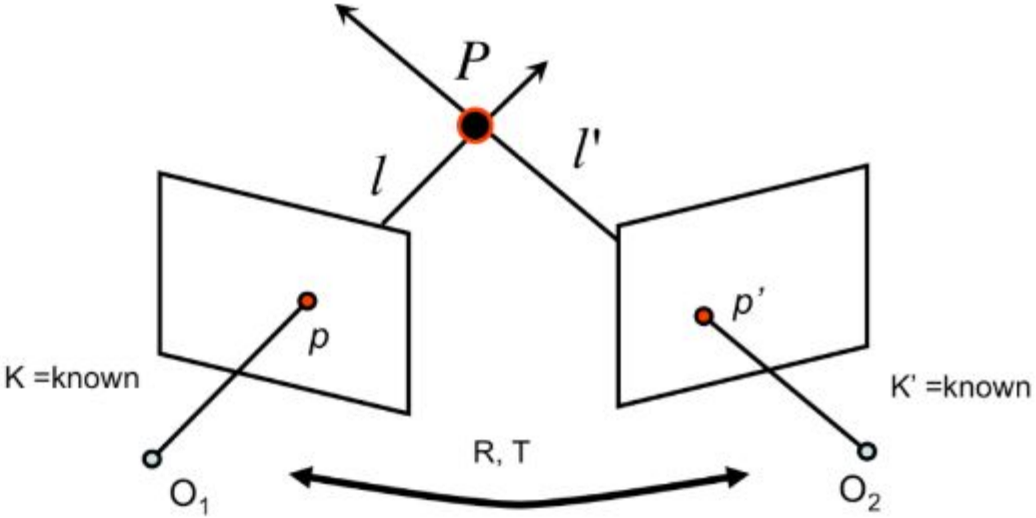
- Introduction.
- Method.
- Schur’s complement trick.

Recap:

- Triangulation:

  Input: Intrinsics, relative orientation b/w frames, projection of a 3D point in frames.

  Output: 3D point coordinates (P).

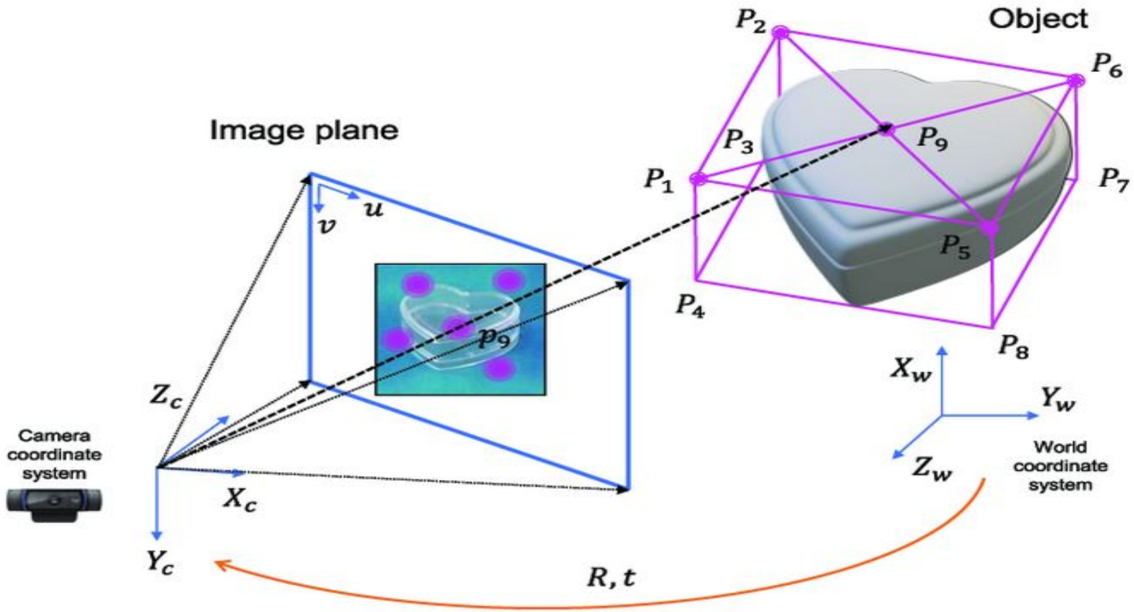


Recap:

- PnP:

  Input: set of correspondences between 3D points of the object and their projections on the image plane.

  Output: relative 6 DoF pose b/w camera and object.



SFM:

Input: Given unordered pair of images

Output: Poses of each of these images along with the structure of the scene.

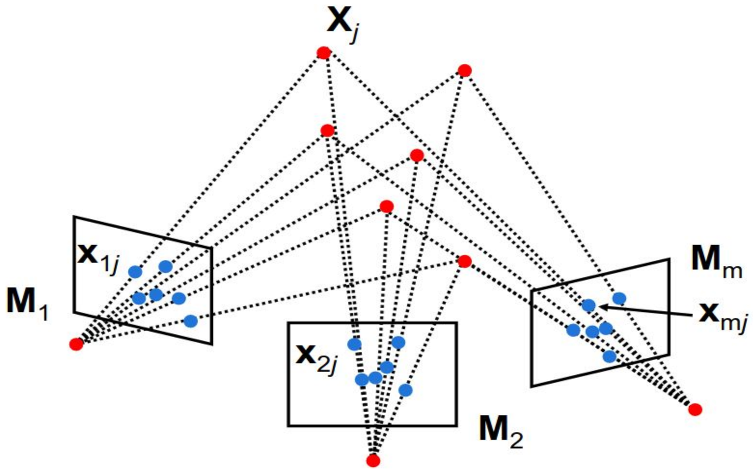
How is this different from SLAM?

SFM Problem Statement:

- We have an unordered collection of M images.

  Notation: (P_i is the proj matrix of ith camera), set of 3d points X_j.
  Note: Each of the 3d points may be visible in one or more cameras.

  Input: x_ij is the projection of X_j on P_i. These x's can be called as observations.
  Output: How can you recover motion of cameras (P_i 's) and structure of the scene (X_j 's).



Ref: http://theia-sfm.org/

Affine SFM:

- Weak perspective proj (orthographic) : Dist from COP to image plane is inf.

$$M = \begin{bmatrix} A & b \\ v & 1 \end{bmatrix}$$

$$x = MX = \begin{bmatrix} & m_1 & & \\ & m_2 & & \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ 1 \end{bmatrix} = \begin{bmatrix} m_1 X \\ m_2 X \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} m_1 X \\ m_2 X \end{bmatrix} = \begin{bmatrix} A & b \end{bmatrix} X = AX + b \longrightarrow \mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

- Unknowns : 8m+3n, equations: 2mn. Where n is no of 3D points , m is no of cameras.

$$\mathbf{x}_{ij} = \mathbf{A}_i \mathbf{X}_j + \mathbf{b}_i, \quad i = 1, \dots, m, \; j = 1, \dots, n$$

Ref: CS231A notes and slides

Tomasi Kanade Factorization for Affine SFM:

Steps:

1. **Data centering**: Centre all the 2D points in every pose.

$$\hat{\mathbf{x}}_{ij} = \mathbf{x}_{ij} - \frac{1}{n}\sum_{k=1}^{n} \mathbf{x}_{ik} = \mathbf{A}_i \mathbf{X}_j + \mathbf{b}_i - \frac{1}{n}\sum_{k=1}^{n} \left(\mathbf{A}_i \mathbf{X}_k + \mathbf{b}_i\right)$$

$$= \mathbf{A}_i \left(\mathbf{X}_j - \frac{1}{n}\sum_{k=1}^{n} \mathbf{X}_k \right) = \mathbf{A}_i \hat{\mathbf{X}}_j$$

Assuming centre of world frame is at centroid of all 3D points.

$$\hat{\mathbf{x}}_{ij} = \mathbf{A}_i \mathbf{X}_j$$

Affine SFM:

- Get the observation matrix x_cap(ij) – (2m*n).

$$\mathbf{D} = \begin{bmatrix} \hat{\mathbf{x}}_{11} & \hat{\mathbf{x}}_{12} & \cdots & \hat{\mathbf{x}}_{1n} \\ \hat{\mathbf{x}}_{21} & \hat{\mathbf{x}}_{22} & \cdots & \hat{\mathbf{x}}_{2n} \\ & & \ddots & \\ \hat{\mathbf{x}}_{m1} & \hat{\mathbf{x}}_{m2} & \cdots & \hat{\mathbf{x}}_{mn} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_m \end{bmatrix}}_{\substack{\text{cameras} \\ (2m \times 3)}} \underbrace{\begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \cdots & \mathbf{X}_n \end{bmatrix}}_{\text{points } (3 \times n)}$$

D = M * S where M is motion matrix and S is structure/shape matrix. How to solve for M, S from D?

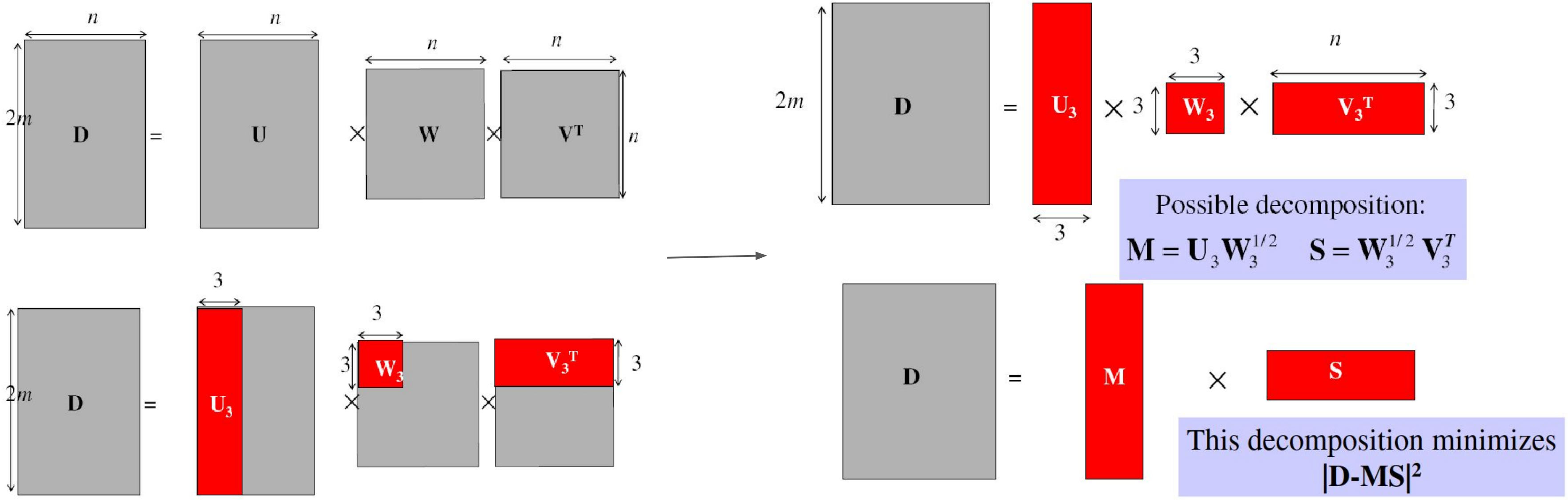
- Find M ,S which minimizes $|\mathbf{D}\text{-}\mathbf{MS}|^2$

Ref: CMU 16.385 slides (lec 12)

Affine SFM….

2. Factorization step:

- Use SVD decomposition as shown below.



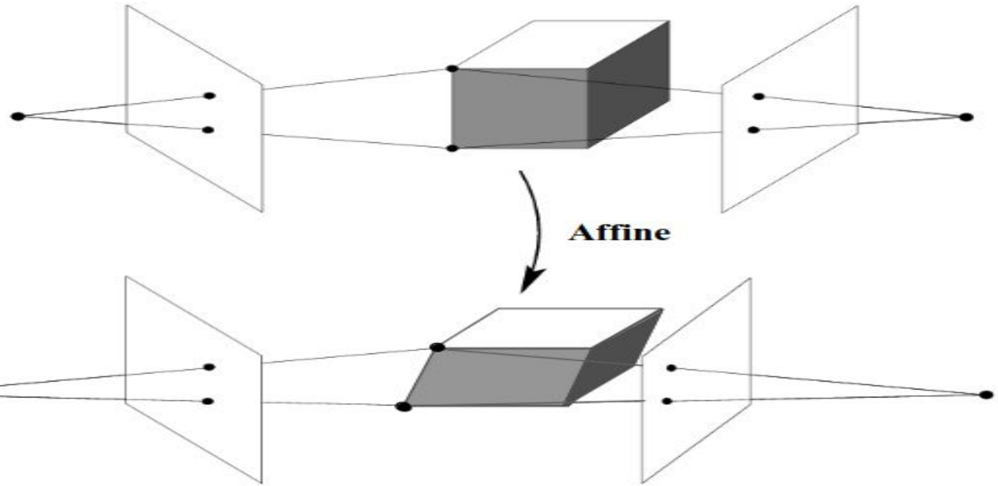
Ref: CMU 16.385 slides (lec 12)

Affine Ambiguity:

- Noticed something in decomposition? (Hint: Multiple solutions for matrices M, S). That’s called ambiguity.

  Take any 3*3 matrix C, we can have below decomposition as well.

$$\mathbf{M} \rightarrow \mathbf{MC},\ \mathbf{S} \rightarrow \mathbf{C^{-1}S}$$

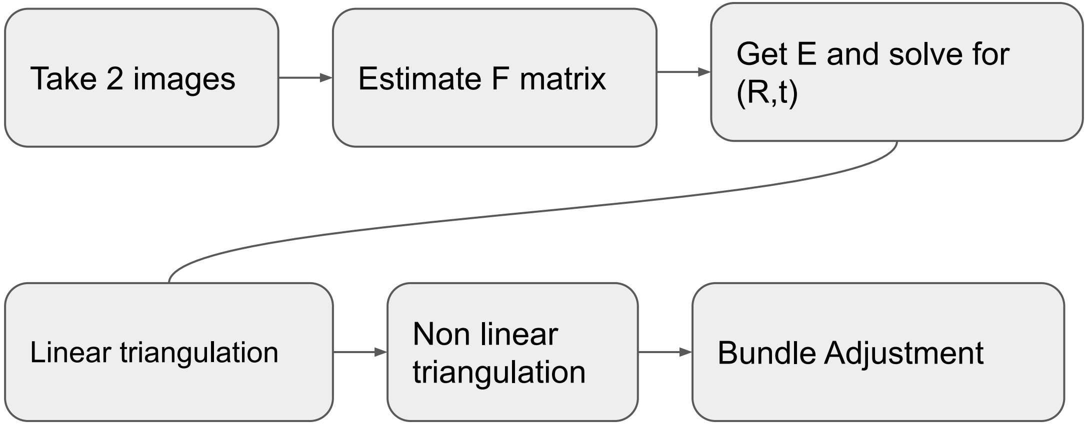


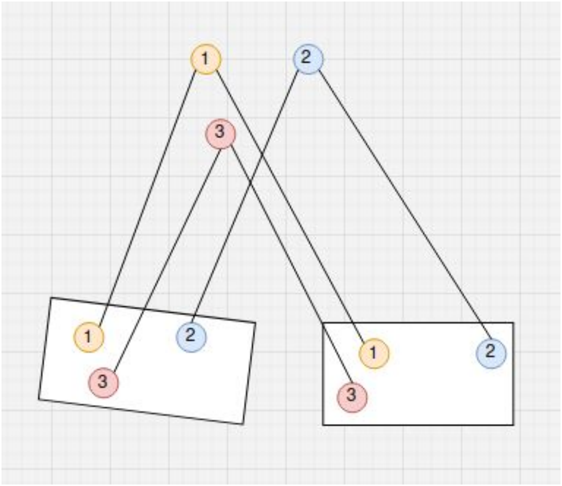
Ref: CMU 16.385 slides (lec 12)

Projective SFM:

- Let’s consider generic case where all cameras as projective. Solved using incremental paradigm.

  Step 1: Start with just 2 images and build sparse structure of scene.

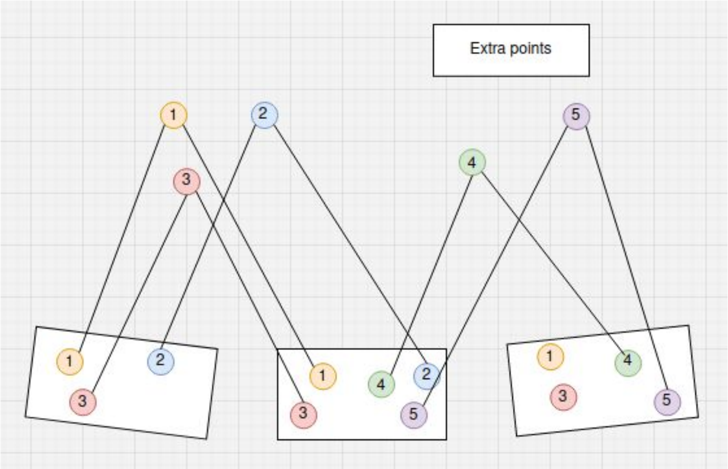




Add more images in SFM..

Step 2: Add images and build structure in incremental fashion.

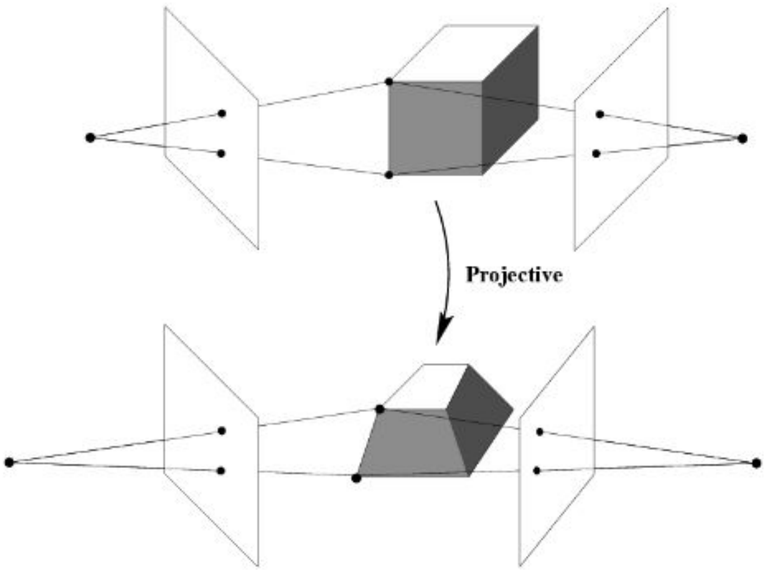
Get 2D-3D correspondences → Estimate pose using PnP → Non linear PnP → Linear triangulation → Non linear triangulation → Bundle Adjustment

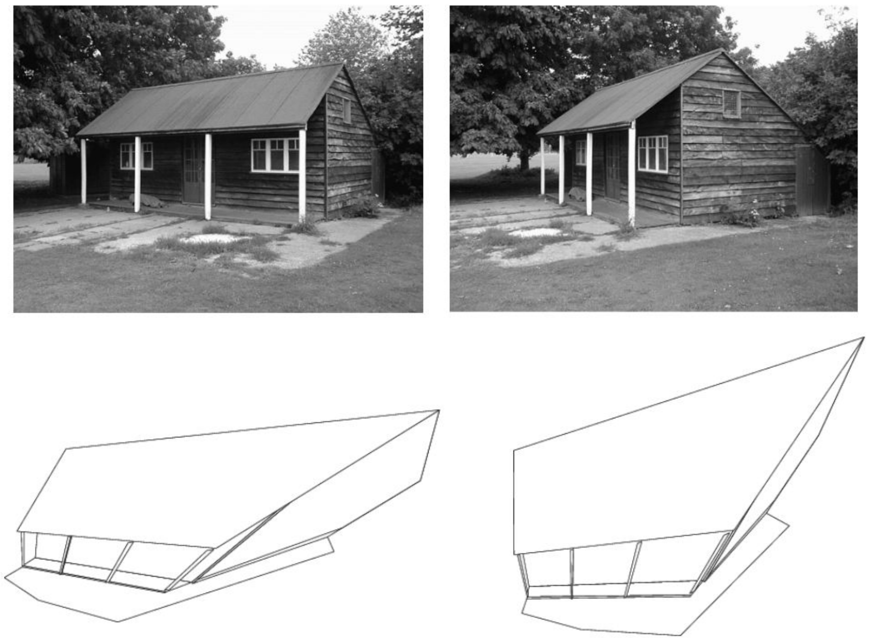


Note: BA above is not performed after every new view is added, instead its done periodically or when you reach a keyframe (first introduced in PTAM). More in SLAM.

Projective ambiguity:

- Reconstruction is upto a projective transformation.





Ref: https://cvgl.stanford.edu/teaching/cs231a_winter1415/lecture/lecture7_perspective_SFM_notes.pdf

1.) Structure from motion

- Problem statement.
- Affine SFM.
- Projective SFM.
- Understand the ambiguities.

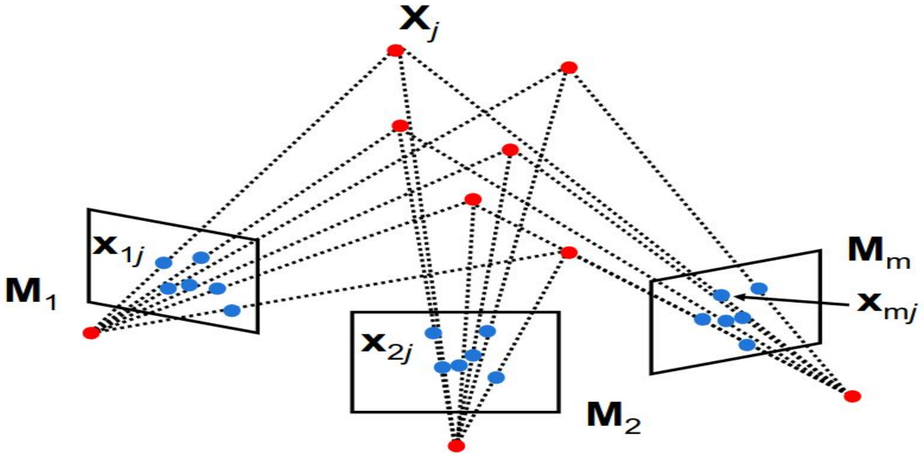
2.) Bundle adjustment.

- Introduction.
- Method.
- Schur’s complement trick.

Bundle adjustment:

- Refine projection matrices and 3D points jointly using iterative optimization algos like LM etc. Why?

- **How**?
  Reproject 3D points on every image and minimize reprojection error by formulating as non linear least squares problem.



Num frames = m
Num points = n

- How many variables to refine/optimize? Let's denote it by vector $\boldsymbol{\phi}$

Bundle adjustment cost function:

- Reprojection error:

$$\text{reprojection error } (R) = \sum_{i=1}^{m} \sum_{j=1}^{n} \| P_i X_j - \mathbf{x}_{ij} \|^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} \| \hat{\mathbf{x}}_{ij} - x_{ij} \|^2$$

To refine: P_i's and X_j's

Known : x_{ij}'s

$$\longrightarrow \mathbf{x}_{ij} = \begin{bmatrix} x_{ij} \\ y_{ij} \end{bmatrix}_{2\times 1}, \quad P_i = \begin{bmatrix} P_{i1}^T \\ P_{i2}^T \\ P_{i3}^T \end{bmatrix}_{3\times 4}$$

$$R = \sum_{i=1}^{m} \sum_{j=1}^{n} \left\{ \left(\frac{P_{i1}^T X_j}{P_{i3}^T X_j} - x_{ij} \right)^2 + \left(\frac{P_{i2}^T X_j}{P_{i3}^T X_j} - y_{ij} \right)^2 \right\} - (1)$$

LM solver recap

- Consider m dimensional $F(x) = [f_1(x), \ldots, f_m(x)]^\top$ fn where x ϵ R

$$\min_x \frac{1}{2}\|F(x)\|^2 .$$

Steps :

1. At every iteration, find correction in x which decreases error.

   We linearize the fn $F(x + \Delta x) \approx F(x) + J(x)\Delta x,$ where $J_{ij}(x) = \partial_j f_i(x)$

   Find correction by minimizing $\min_{\Delta x} \frac{1}{2}\|J(x)\Delta x + F(x)\|^2$

2. Update x :

$$\Delta x \longleftarrow (J^T J + \lambda I)^{-1} J^T r$$

$$x \longleftarrow x + \Delta x$$

Ref: Bundle adjustment in Large (Sameer El al.)

Bundle adjustment …

- Step 1: Construct residual vector

Get residual vector for each pose:-

$$r_i = \begin{bmatrix} \hat{X}_{i1} - X_{i1} \\ \hat{X}_{i2} - X_{i2} \\ \vdots \\ \hat{X}_{in} - X_{in} \end{bmatrix} = \begin{bmatrix} \hat{x}_{i1} - x_{i1} \\ \hat{y}_{i1} - y_{i1} \\ \vdots \\ \hat{x}_{in} - x_{in} \\ \hat{y}_{in} - y_{in} \end{bmatrix}_{2n \times 1}$$

Net residual vector (r) $= \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_m \end{bmatrix}_{2mn \times 1}$ $\quad (R = r^T r)$

Bundle adjustment …

- Step 2: Compute Motion Jacobian.

a) Motion Jacobian :-

$$J_M = \frac{\partial r}{\partial P} = \begin{bmatrix} \frac{\partial r}{\partial P_1} & \frac{\partial r}{\partial P_2} & \cdots & \frac{\partial r}{\partial P_m} \end{bmatrix}_{2mn \times 12m}$$

where

$$\frac{\partial r}{\partial P_i} = \begin{bmatrix} \frac{\partial r_1}{\partial P_i} \\ \frac{\partial r_2}{\partial P_i} \\ \vdots \\ \frac{\partial r_m}{\partial P_i} \end{bmatrix}_{2mn \times 12}$$

Observe that

$$\frac{\partial r_k}{\partial P_i} = \left\{ \begin{array}{ll} 0, & \text{if } k \neq i \\ = \begin{bmatrix} \frac{\partial \hat{x}_{i1}}{\partial P_i} \\ \frac{\partial \hat{y}_{i1}}{\partial P_i} \\ \vdots \\ \frac{\partial \hat{x}_{in}}{\partial P_i} \\ \frac{\partial \hat{y}_{in}}{\partial P_i} \end{bmatrix}_{2n \times 12} & \text{if } k = i \end{array} \right\}_{2n \times 12}$$

Bundle adjustment …

$$\Rightarrow J_M = \begin{bmatrix} \frac{\partial \hat{x}_{11}}{\partial P_1} & 0 & \cdots & 0 \\ \frac{\partial \hat{y}_{11}}{\partial P_1} & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ \frac{\partial \hat{x}_{1n}}{\partial P_1} & \vdots & & \vdots \\ \frac{\partial \hat{y}_{1n}}{\partial P_1} & 0 & \cdots & 0 \\ 0 & \frac{\partial \hat{x}_{21}}{\partial P_2} & & \vdots \\ \vdots & \frac{\partial \hat{y}_{21}}{\partial P_2} & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \frac{\partial \hat{x}_{2n}}{\partial P_2} & & \vdots \\ \vdots & \frac{\partial \hat{y}_{2n}}{\partial P_2} & & \vdots \\ \vdots & 0 & & \vdots \\ \vdots & \vdots & & \frac{\partial \hat{x}_{m1}}{\partial P_m} \\ \vdots & \vdots & & \frac{\partial \hat{y}_{m1}}{\partial P_m} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & & \frac{\partial \hat{x}_{mn}}{\partial P_m} \\ 0 & 0 & & \frac{\partial \hat{y}_{mn}}{\partial P_m} \end{bmatrix}$$

2n×12

(1×12)

2n×12

2n×12

Motion Jacobian

Bundle adjustment …

- Step 3: Compute Structure Jacobian.

b) Structure Jacobian (J_S):

$$J_S = \frac{\partial r}{\partial X} = \begin{bmatrix} \frac{\partial r}{\partial X_1} & \frac{\partial r}{\partial X_2} & \cdots & \frac{\partial r}{\partial X_n} \end{bmatrix}_{2mn \times 3n}$$

where
$$\frac{\partial r}{\partial X_i} = \begin{bmatrix} \frac{\partial r_1}{\partial X_i} \\ \vdots \\ \frac{\partial r_m}{\partial X_i} \end{bmatrix}_{2mn \times 3}, \quad \frac{\partial r_k}{\partial X_i} = \begin{bmatrix} \frac{\partial \hat{x}_{k1}}{\partial X_i} \\ \frac{\partial \hat{y}_{k1}}{\partial X_i} \\ \vdots \\ \frac{\partial \hat{x}_{kn}}{\partial X_i} \\ \frac{\partial \hat{y}_{kn}}{\partial X_i} \end{bmatrix}_{2n \times 3}$$

$$\boxed{X_i = \begin{bmatrix} x_i^w \\ y_i^w \\ z_i^w \end{bmatrix}}$$

$$\Rightarrow \frac{\partial r_k}{\partial X_i} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \frac{\partial \hat{x}_{ki}}{\partial X_i} \\ \frac{\partial \hat{y}_{ki}}{\partial X_i} \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{2n \times 3} \quad \forall k = 1 \text{ to } m$$

Bundle adjustment …

$$
\begin{array}{cccccl}
\frac{\partial \hat{x}_{11}}{\partial x_1} & 0 & - & - & 0 & \rightarrow 1\times 3 \\
\frac{\partial \hat{y}_{11}}{\partial x_1} & 0 & - & - & 0 & \\
0 & \frac{\partial \hat{x}_{12}}{\partial x_2} & - & - & 0 & \\
0 & \frac{\partial \hat{y}_{12}}{\partial x_2} & - & - & 0 & \\
\vdots & \vdots & & & \vdots & \\
0 & 0 & - & - & \frac{\partial \hat{x}_{1n}}{\partial x_n} & \\
0 & 0 & - & - & \frac{\partial \hat{y}_{1n}}{\partial x_n} & \\
\frac{\partial \hat{x}_{21}}{\partial x_1} & 0 & - & - & 0 & \\
\frac{\partial \hat{y}_{21}}{\partial x_1} & 0 & - & - & 0 & \\
0 & \frac{\partial \hat{x}_{22}}{\partial x_2} & - & - & 0 & \\
0 & \frac{\partial \hat{y}_{22}}{\partial x_2} & - & - & 0 & \\
\vdots & \vdots & & & \vdots & \\
0 & 0 & - & - & \frac{\partial \hat{x}_{2n}}{\partial x_n} & \\
0 & 0 & - & - & \frac{\partial \hat{y}_{2n}}{\partial x_n} & \\
\vdots & \vdots & & & \vdots & \\
\frac{\partial \hat{x}_{m1}}{\partial x_1} & 0 & - & - & 0 & \\
\frac{\partial \hat{y}_{m1}}{\partial x_1} & 0 & - & - & 0 & \\
0 & \frac{\partial \hat{x}_{m2}}{\partial x_2} & - & - & 0 & \\
0 & \frac{\partial \hat{y}_{m2}}{\partial x_2} & - & - & 0 & \\
\vdots & \vdots & & & \vdots & \\
0 & 0 & - & - & \frac{\partial \hat{x}_{mn}}{\partial x_n} & \\
0 & 0 & - & - & \frac{\partial \hat{y}_{mn}}{\partial x_n} &
\end{array}
$$

Structure Jacobian

Bundle adjustment …

- Step 4: Update rule.

Combined Jacobian $J = [\, J_M \mid J_S \,]$

J_M: $2mn \times 12m$ $\qquad$ J_S: $2mn \times 3n$

$J \rightarrow 2mn \times (12m + 3n)$

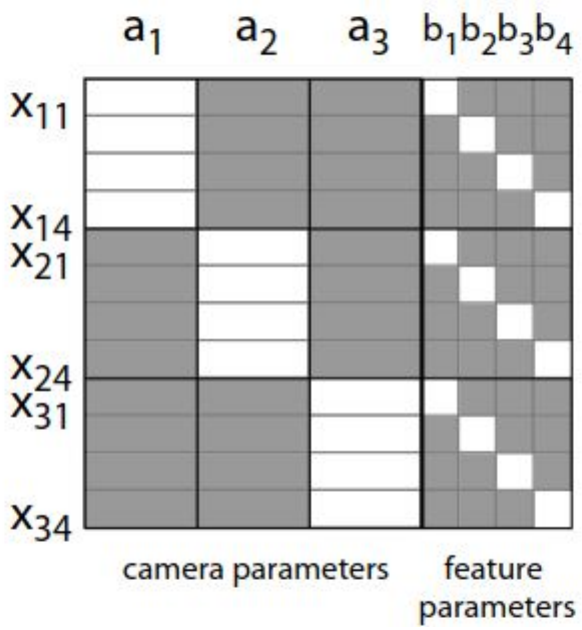
Using LM solver, we can get update

i.e $\delta \leftarrow \underbrace{(J^T J + \lambda I)^{-1} J^T r}_{(12m+3n)\times 1}$

a. $\boldsymbol{\delta}$ to be computed at every iteration of LM algorithm. (Also called as normal equation).
b. $\boldsymbol{\phi} \longleftarrow \boldsymbol{\phi} - \boldsymbol{\delta}$ (update rule)

Sparse Bundle adjustment

- What's the complexity of the above update rule?
- How to exploit sparse structure of the Jacobians?



Source: **Towards using sparse bundle adjustment for robust stereo odometry in outdoor terrain**

Schur complement trick

Let $J = [J_M \mid J_S] = [A \mid B]$

$$\rightarrow \delta = \begin{pmatrix} \delta_a \\ \delta_b \end{pmatrix}$$

$a \rightarrow$ motion

$b \rightarrow$ structure

$$\Rightarrow J^T J = \begin{bmatrix} A^T \\ B^T \end{bmatrix} \begin{bmatrix} A & B \end{bmatrix} = \left[\begin{array}{c|c} A^T A & A^T B \\ \hline B^T A & B^T B \end{array}\right]$$

Normal eq'n $\rightarrow$

$$\begin{bmatrix} A^T A & A^T B \\ B^T A & B^T B \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_b \end{bmatrix} = \begin{bmatrix} A^T r \\ B^T r \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} U & W \\ W^T & V \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_b \end{bmatrix} = \begin{pmatrix} \varepsilon_a \\ \varepsilon_b \end{pmatrix}$$

Ref: Towards using sparse bundle adjustment for robust stereo odometry in outdoor terrain

Schur complement trick

Pre Multiply by $\begin{bmatrix} I & -WV^{-1} \\ 0 & I \end{bmatrix}$

$$\Rightarrow \begin{bmatrix} I & -WV^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} U & W \\ W^T & V \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_b \end{bmatrix} = \begin{bmatrix} I & -WV^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} \varepsilon_a \\ \varepsilon_b \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} U - WV^{-1}W^T & 0 \\ W^T & V \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_b \end{bmatrix} = \begin{bmatrix} \varepsilon_a - WV^{-1}\varepsilon_b \\ \varepsilon_b \end{bmatrix}$$

$$\Rightarrow \underbrace{\left(U - WV^{-1}W^T\right)}_{(12m \times 12m) \ll ((12m+3n) \times (12m+3n))} \delta_a = \varepsilon_a - WV^{-1}\varepsilon_b \,, \quad V\delta_b = \varepsilon_b - W^T\delta_a$$

Ref: NUS 3D Vision course