COMPUTER VISION AND PHOTOGRAMMETRY

MOTION AND STRUCTURE REFINEMENT

3D MODELLING FROM IMAGES PIPELINE

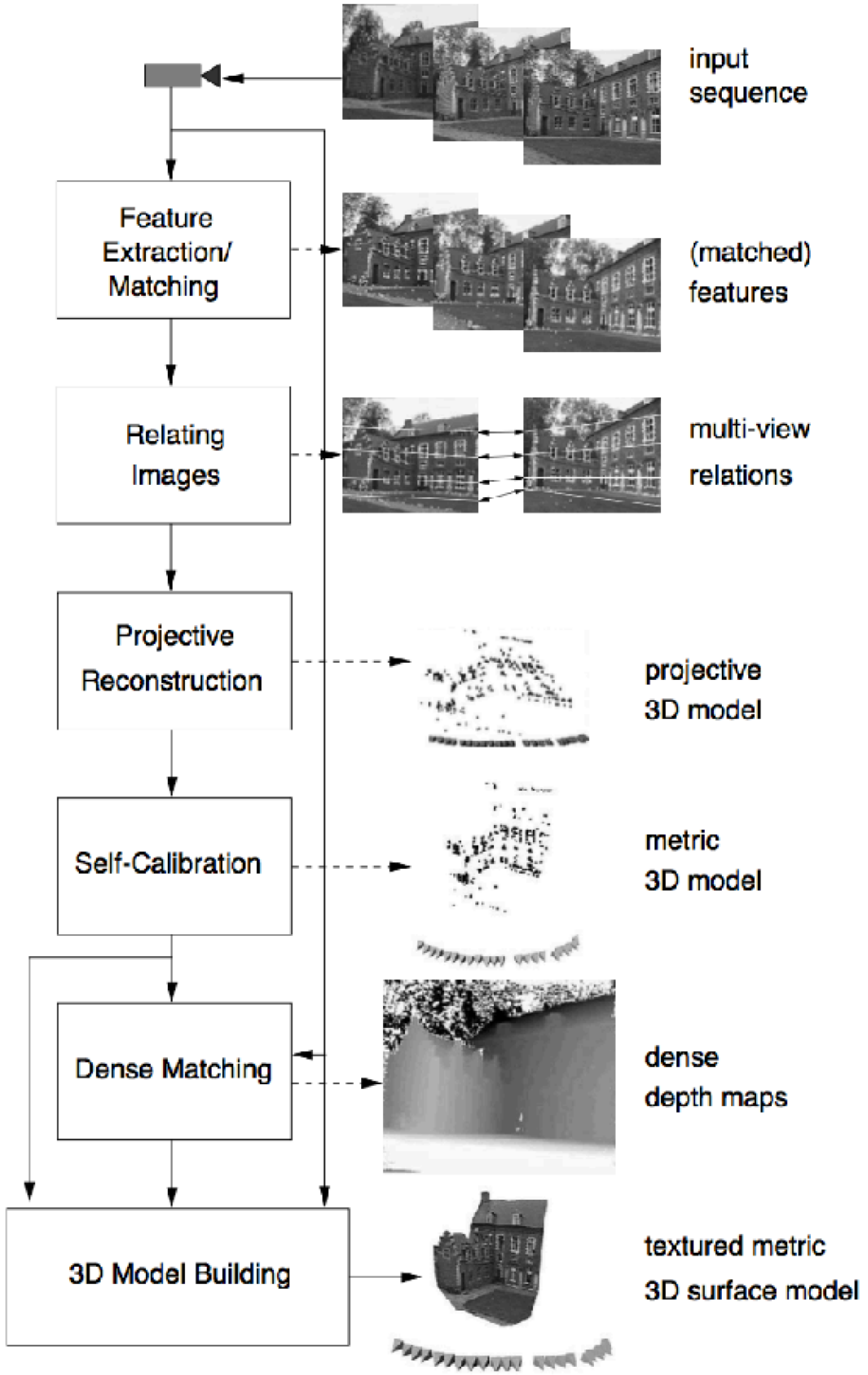


Figure 1.7: Overview of the presented approach for 3D modeling from images

SEQUENTIAL STRUCTURE AND MOTION RECOVERY

Initialize structure and motion from 2 views

For each additional view

- Determine pose
- Refine and extend structure

INITIAL STRUCTURE AND MOTION

Epipolar geometry ↔ Projective calibration

$$\mathbf{m}_2^{\mathsf{T}}\mathbf{F}\mathbf{m}_1 = 0 \qquad \mathbf{P}_1 = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}$$

$$\mathbf{P}_2 = \begin{bmatrix} [\mathbf{e}]_\times \mathbf{F} + \mathbf{e}\mathbf{a}^{\mathsf{T}} & \mathbf{e} \end{bmatrix}$$

compatible with F

Yields correct projective camera setup

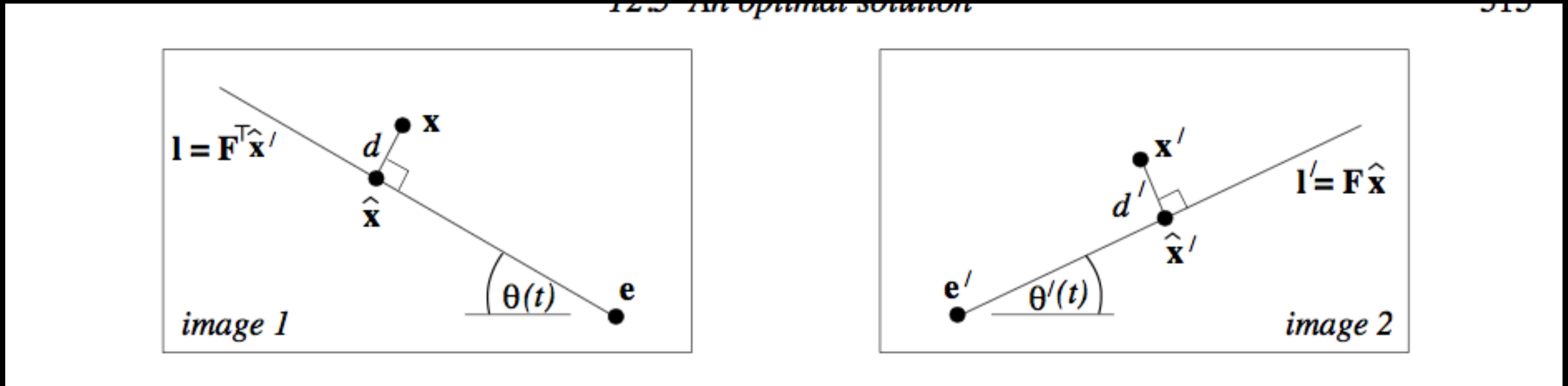
(Faugeras´92,Hartley´92)

INITIAL STRUCTURE AND MOTION

- The first step consists of selecting two views that are suited for initializing the sequential structure and motion computation.
- On the one hand it is important that sufficient features are matched between these views, on the other hand the views should not be too close to each other so that the initial structure is well-conditioned.
- In practice the selection of the initial frame can be done by maximizing the product of the number of matches and a image-based distance based on the planar tomography between images

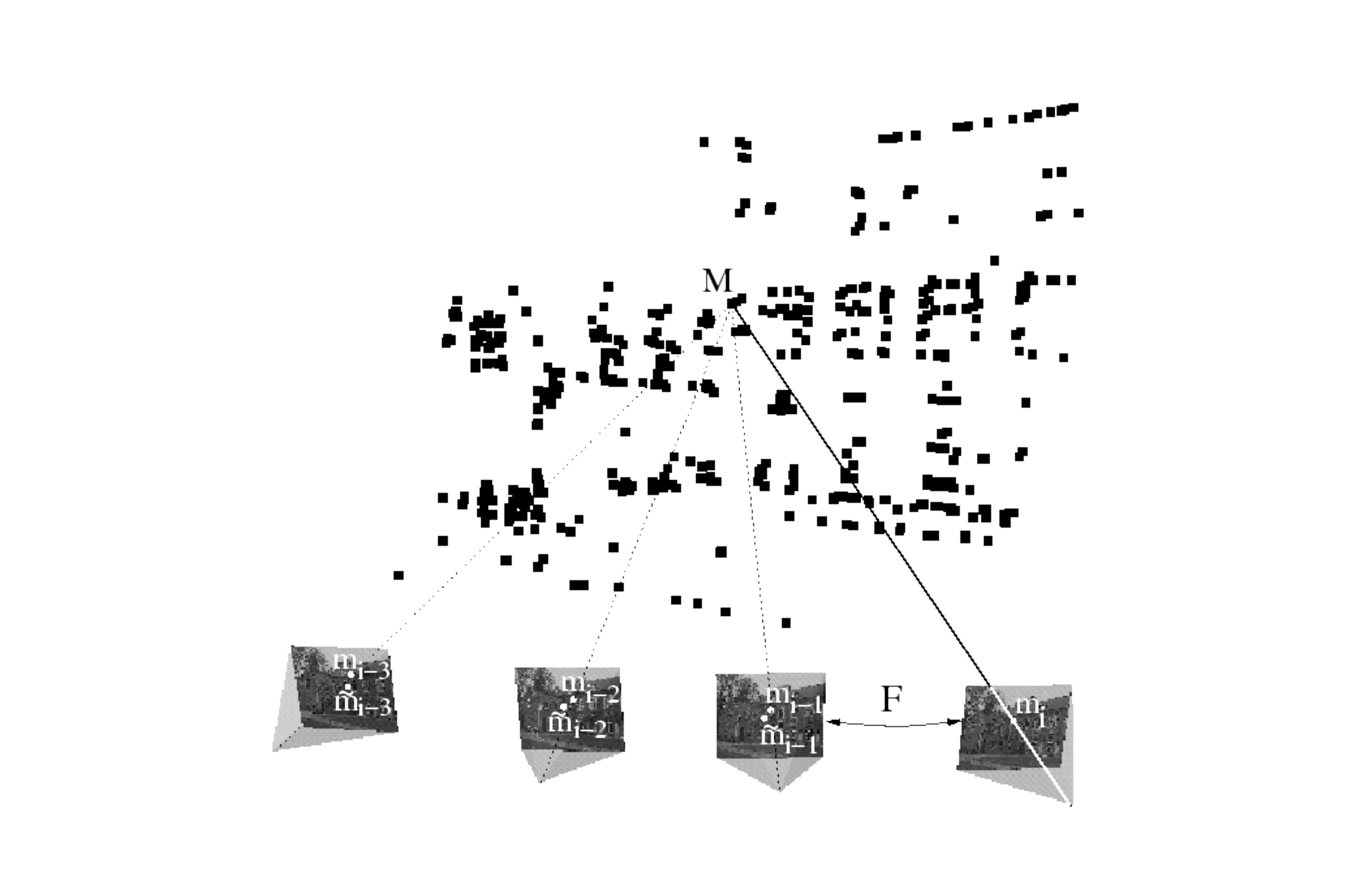
INITIAL STRUCTURE AND MOTION

- Homogeneous method (seen earlier)
- One can pre-adjust points so they satisfy the epipolar constraint.



$$d(\mathbf{x}, \mathbf{l})^2 + d(\mathbf{x}', \mathbf{l}')^2$$

DETERMINE POSE TOWARDS EXISTING STRUCTURE



Compute $\mathbf{P}_{i+1}$ using robust approach (6-point RANSAC)
Extend and refine reconstruction

COMPUTE P WITH 6-POINT RANSAC

- Generate hypothesis using 6 points

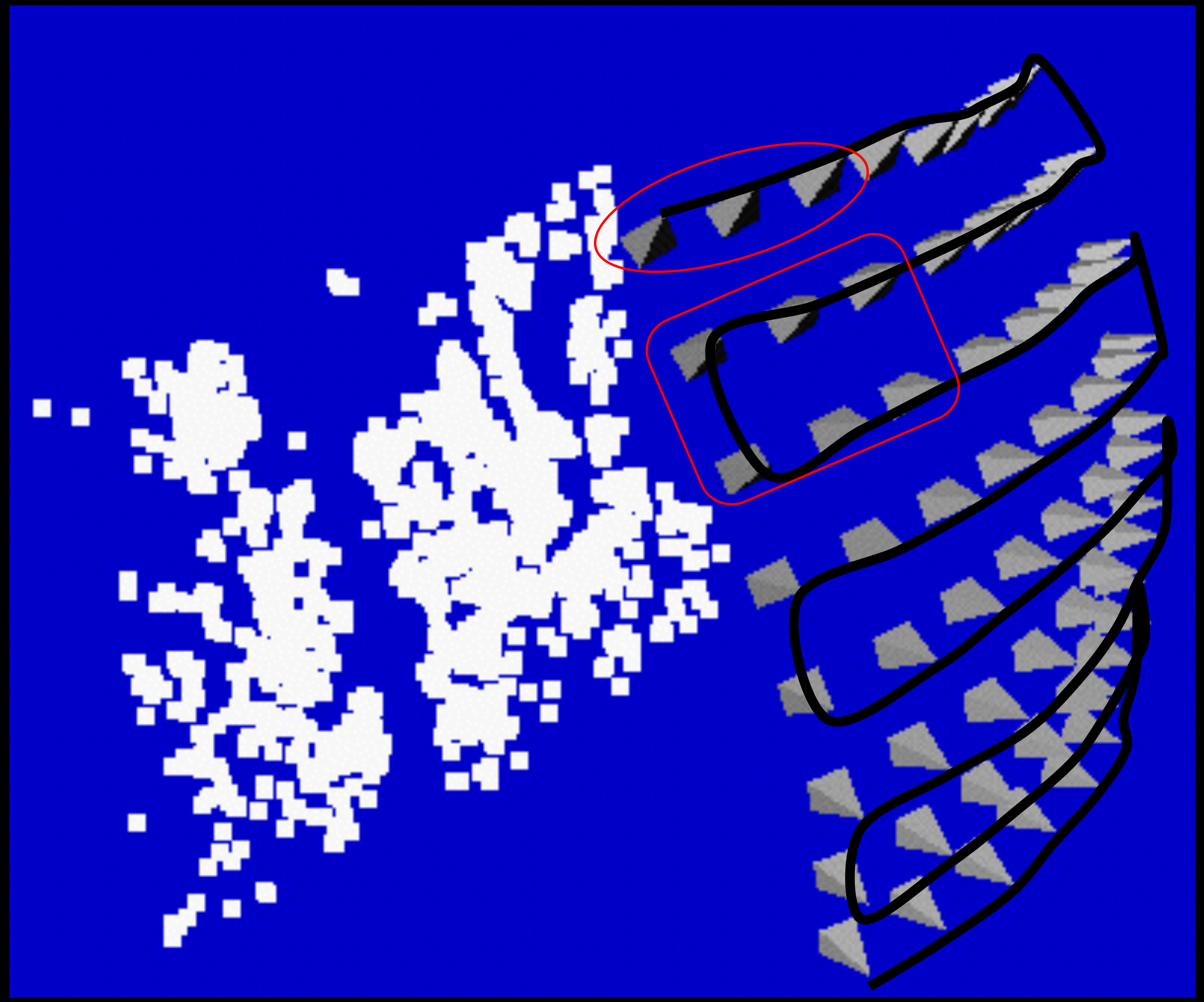
$$\begin{bmatrix} \mathbf{0}^\top & -w_i \mathbf{X}_i{}^\top & y_i \mathbf{X}_i{}^\top \\ w_i \mathbf{X}_i{}^\top & \mathbf{0}^\top & -x_i \mathbf{X}_i{}^\top \end{bmatrix} \begin{pmatrix} \mathbf{P}^1 \\ \mathbf{P}^2 \\ \mathbf{P}^3 \end{pmatrix} = 0$$

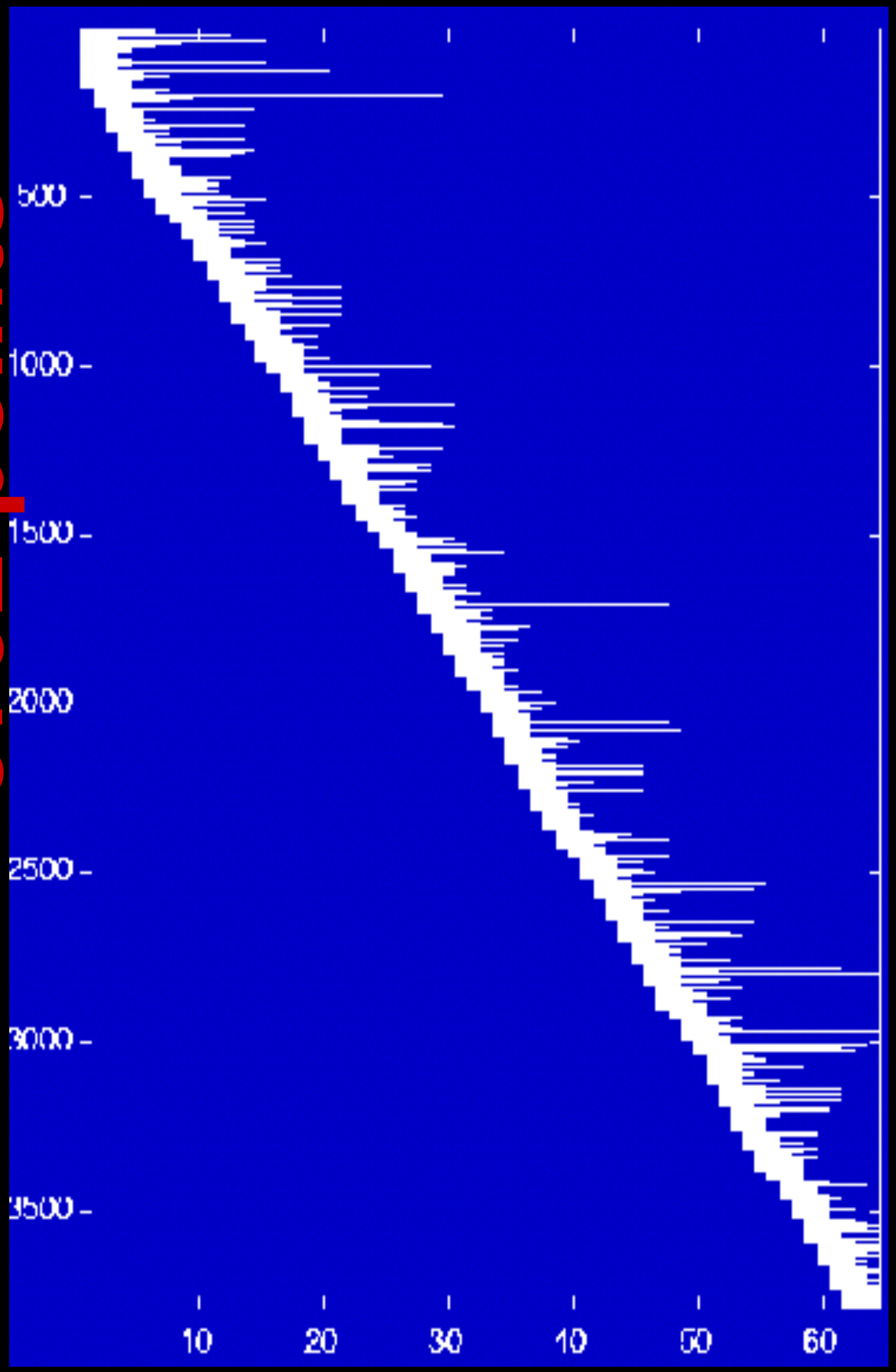
- Count inliers
  - Re-projection error $d\left(\mathrm{P}_i \mathrm{X}\left(\mathrm{x}_1, \ldots, \mathrm{x}_{i-1}, \mathrm{x}_i\right), \mathrm{x}_i\right) < t$

- Expensive testing? Abort early if not promising
  - Also verify at random, abort if e.g. P(wrong)>0.95

(Chum and Matas, BMVC’02)

NON-SEQUENTIAL IMAGE COLLECTIONS





Problem:
Features are lost and reinitialized as new features

Solution:
Match with other *close* views

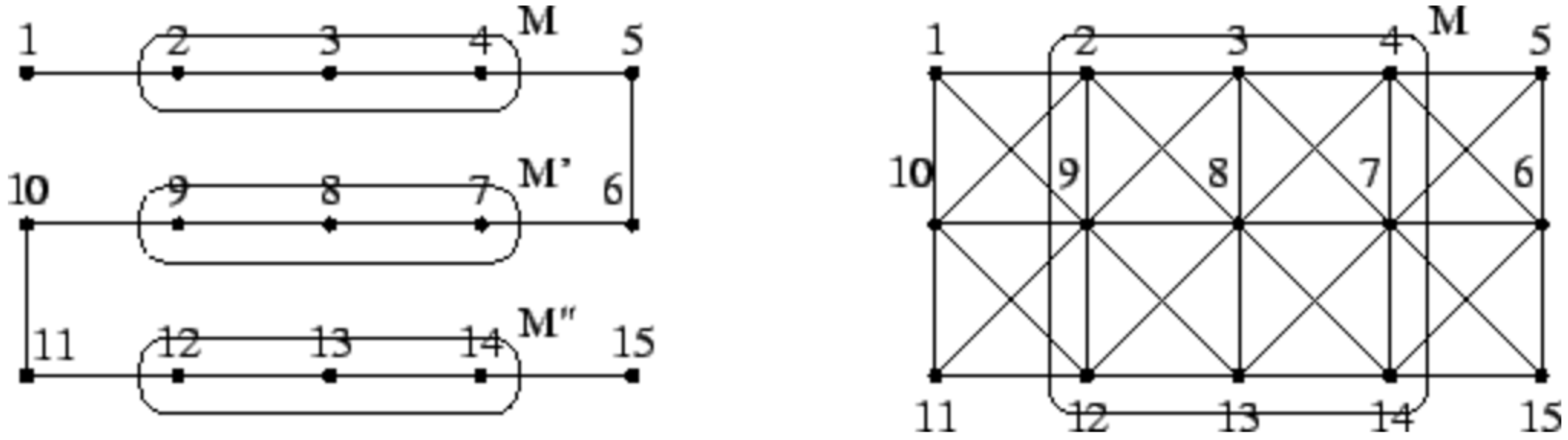


Figure 5.2: *Sequential approach (left) and extended approach (right). In the traditional scheme view 8 would be matched with view 7 and 9 only. A point* `M` *which would be visible in views 2,3,4,7,8,9,12,13 and 14 would therefore result in 3 independently reconstructed points. With the extended approach only one point will be instantiated. It is clear that this results in a higher accuracy for the reconstructed point while it also dramatically reduces the accumulation of calibration errors.*

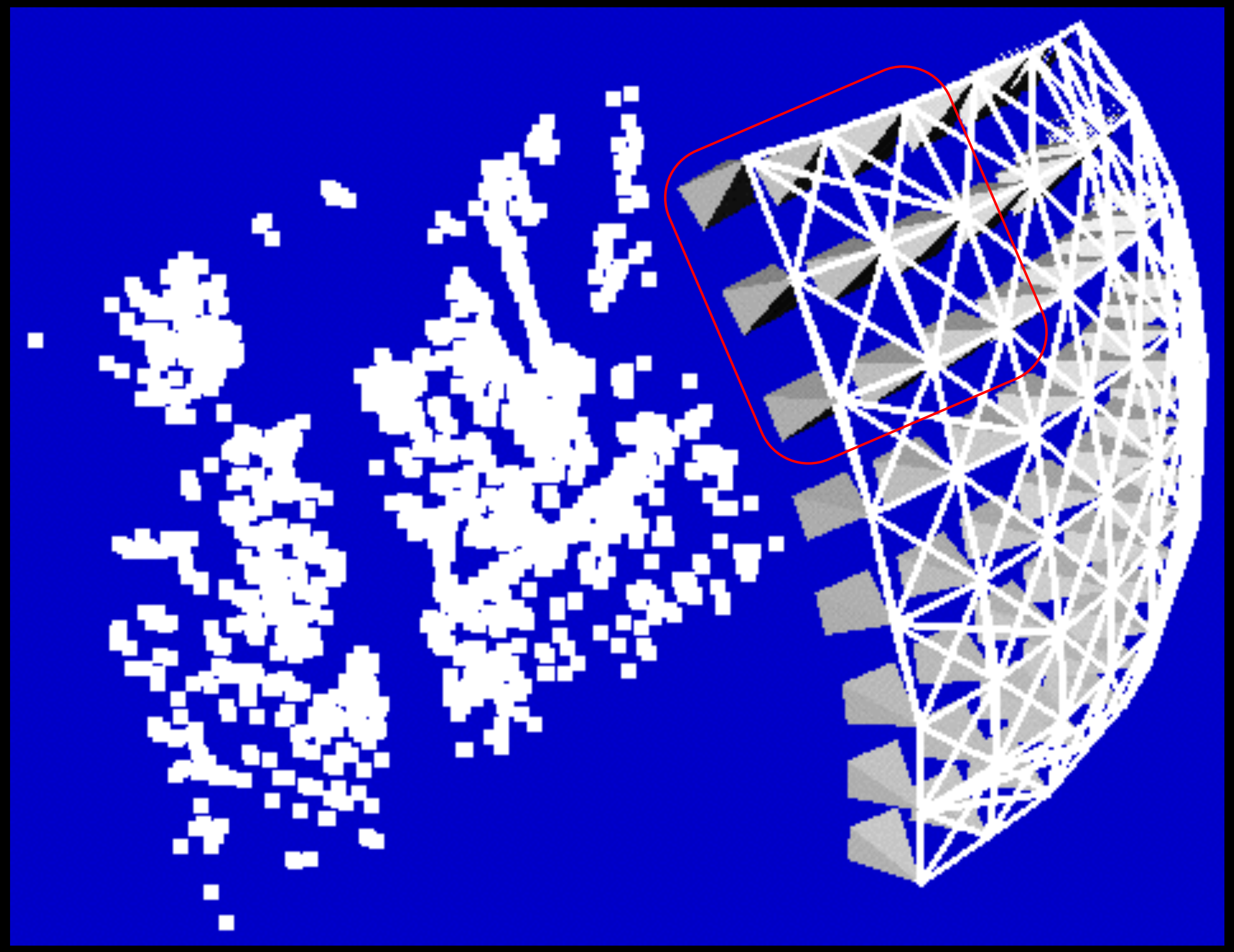
DETERMINING CLOSE VIEWS

- If viewpoints are close then most image changes can be modelled through a planar homography
- Qualitative distance measure is obtained by looking at the residual error on the best possible planar homography

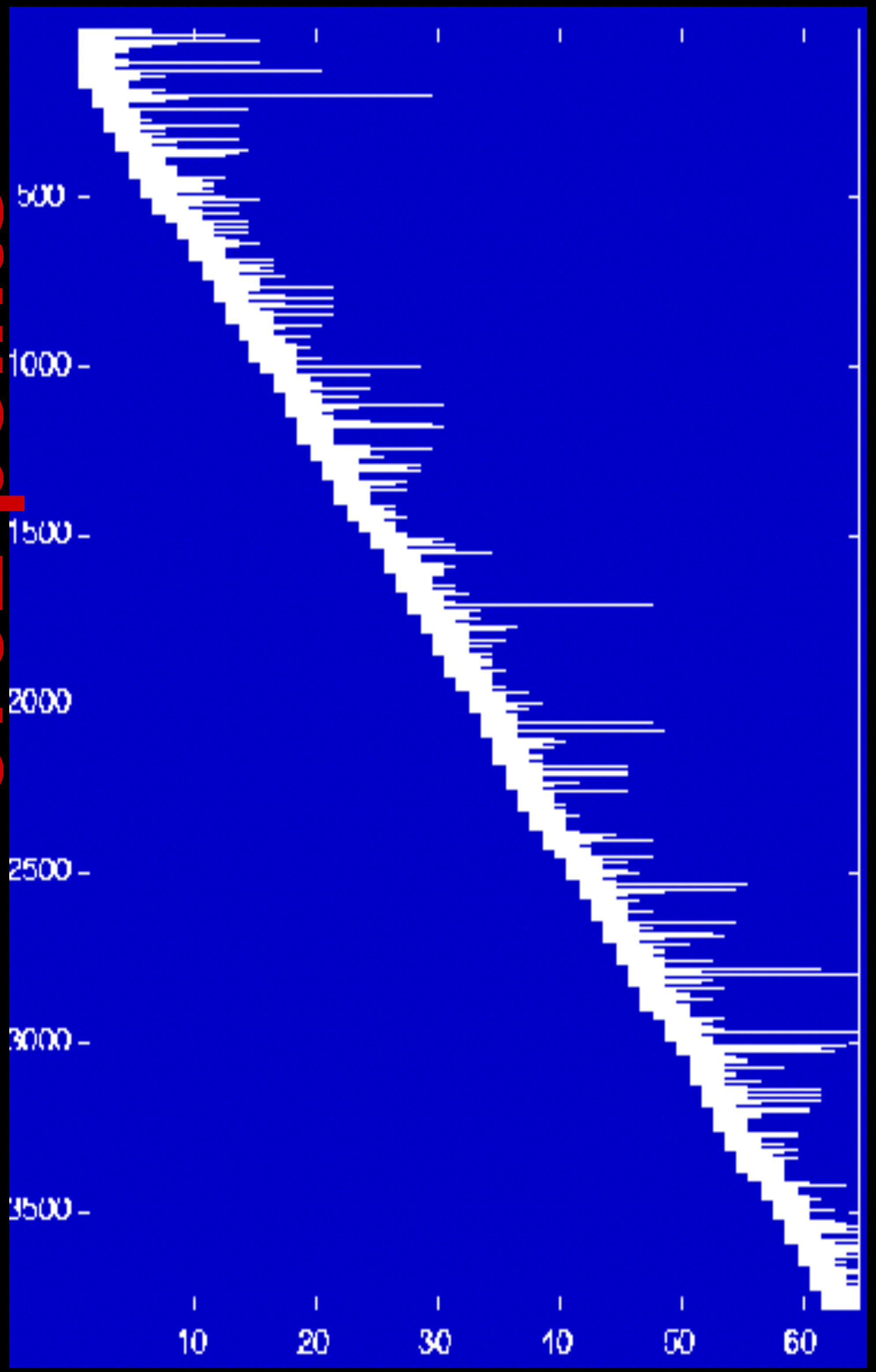
$$\text{Distance} = \min \operatorname{median} D\left(\mathrm{Hm}, \mathrm{m}'\right)$$

$$\mathbf{H} = [\mathbf{e}]_\times \mathbf{F} + \mathbf{e}\mathbf{a}_{min}^\top \text{ with } \mathbf{a}_{min} = \operatorname{argmin}_{\mathbf{a}} \sum_i D(([\mathbf{e}]_\times \mathbf{F} + \mathbf{e}\mathbf{a}^\top)\mathbf{m}_i, \mathbf{m}_i')^2$$

NON-SEQUENTIAL IMAGE COLLECTIONS (2)



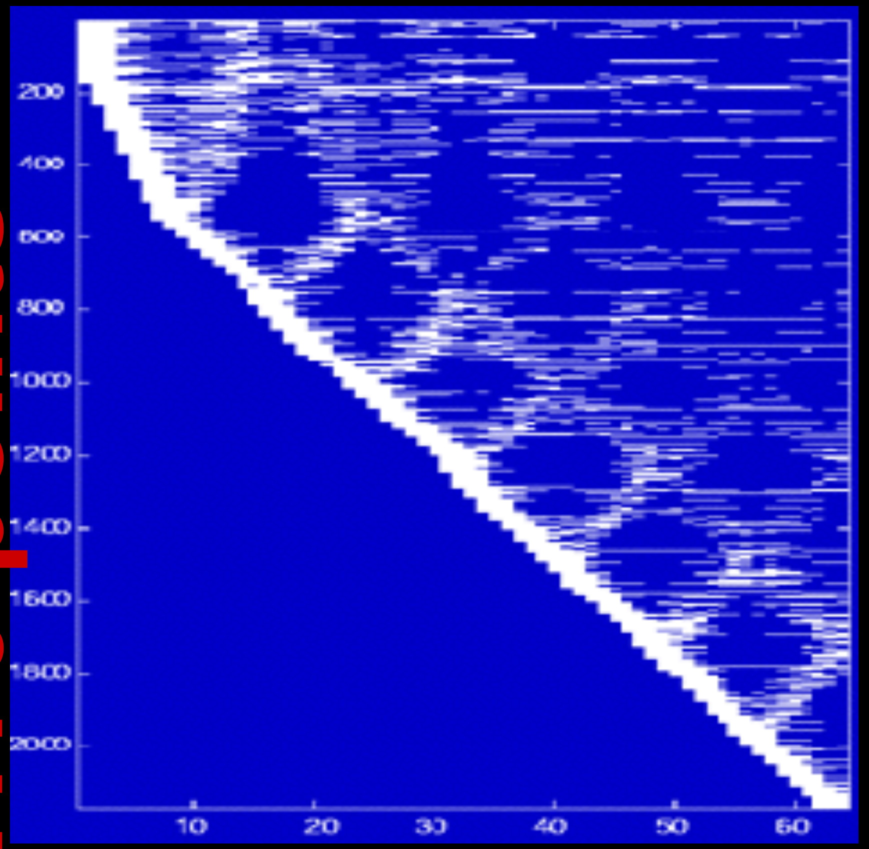
3792 points



4.8im/pt

64 images

2170 points



9.8im/pt

64 images

REFINING GEOMETRY

- By solving this system of equations through SVD a normalized homogeneous point is automatically obtained. If a 3D point is not observed the position is not updated

$$\frac{1}{P_3\tilde{M}}\begin{bmatrix} P_3 x - P_1 \\ P_3 y - P_2 \end{bmatrix} M = 0$$

Step 1. Match or track points over the whole image sequence.

Step 2. Initialize the structure and motion recovery

- step 2.1. Select two views that are suited for initialization.
- step 2.2. Relate these views by computing the two view geometry.
- step 2.3. Set up the initial frame.
- step 2.4. Reconstruct the initial structure.

Step 3. For every additional view

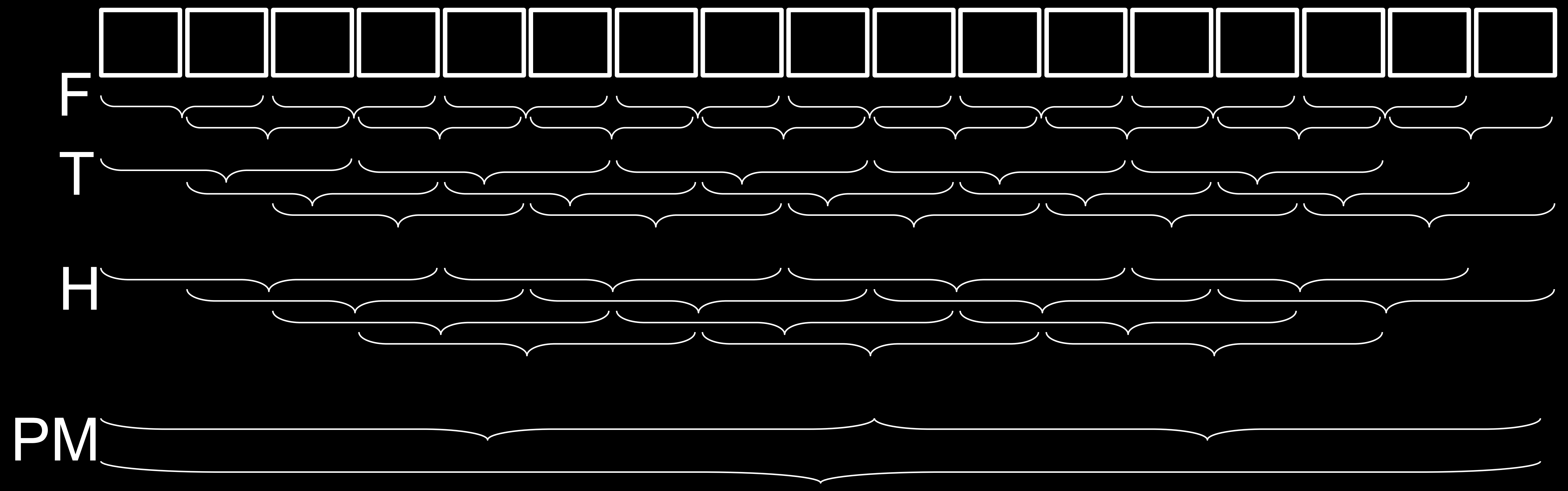
- step 3.1. Infer matches to the structure and compute the camera pose using a robust algorithm.
- step 3.2. Refine the existing structure.
- step 3.3. (optional) For already computed views which are "close"
  - 3.4.1. Relate this view with the current view by finding feature matches and computing the two view geometry.
  - 3.4.2. Infer new matches to the structure based on the computed matches and add these to the list used in step 3.1.

  Refine the pose from all the matches using a robust algorithm.
- step 3.5. Initialize new structure points.

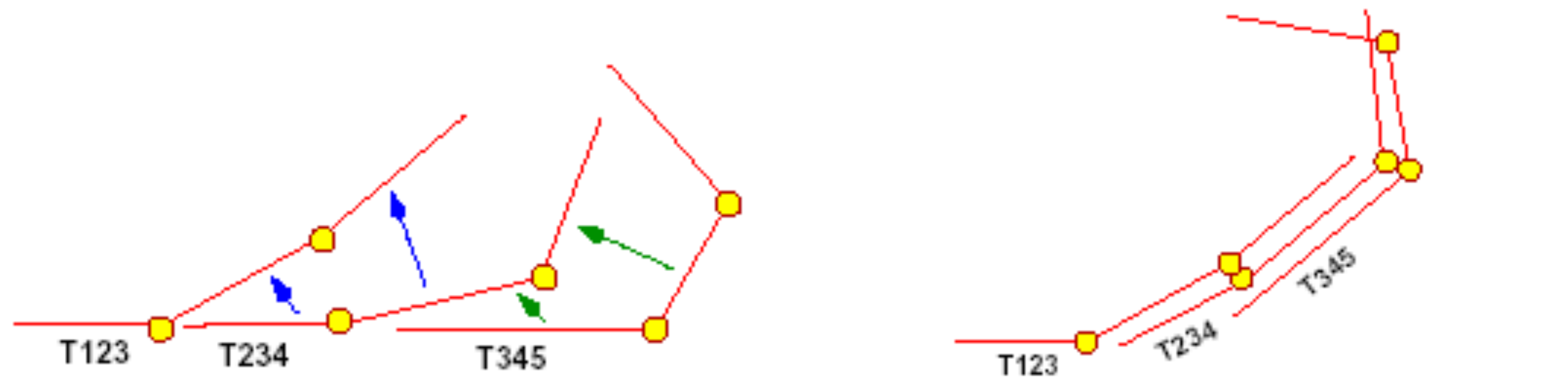
Step 4. Refine the structure and motion through bundle adjustment.

HIERARCHICAL STRUCTURE AND MOTION RECOVERY

- Compute 2-view
- Compute 3-view
- Stitch 3-view reconstructions
- Merge and refine reconstruction



STITCHING 3-VIEW RECONSTRUCTIONS



- Different possibilities
- 1. Align (P2,P3) with (P′1,P′2)
- 2. Align X,X′ (and C′C′)
- 3. Minimize reproj. error
- 4. MLE (merge)

$$\arg\min_{\mathrm{H}} d_A\left(\mathrm{P}_2, \mathrm{P}'_1\mathrm{H}^{-1}\right) + d_A\left(\mathrm{P}_3, \mathrm{P}'_2\mathrm{H}^{-1}\right)$$

$$\arg\min_{\mathrm{H}} \sum_j d_A\left(\mathrm{X}_j, \mathrm{HX}'_j\right)$$

$$\arg\min_{\mathrm{H}} \sum_j d\left(\mathrm{PH}^{-1}\mathrm{X}'_j, \mathrm{x}_j\right) + \sum_j d\left(\mathrm{P}'\mathrm{HX}_j, \mathrm{x}'_j\right)$$

$$\arg\min_{\mathrm{P,X}} \sum_j d\left(\mathrm{PX}_j, \mathrm{x}_j\right)$$

BUNDLE ADJUSTMENT

PROJECTIVE OR METRIC RECONSTRUCTION

- Measurements are noisy
- Estimations introduce errors
- Refining Structure (3D points) and Motion (camera matrices)

REFINING STRUCTURE AND MOTION

- Minimize reprojection error

$$\min_{\hat{P}_k, \hat{M}_i} \sum_{k=1}^{m} \sum_{i=1}^{n} D\left(m_{ki}, \hat{P}_k \hat{M}_i\right)^2$$

  - Maximum Likelyhood Estimation (if error zero-mean Gaussian noise)
  - Huge problem but can be solved efficiently (Bundle adjustment)
  - For a typical sequence of 20 views and 2000 points, a minimization problem in more than 6000 variables has to be solved

NON-LINEAR LEAST-SQUARES

$$X = f(P)$$

$$\underset{P}{\operatorname{argmin}} \left\| X - f(P) \right\|$$

- P parameter vector, X measure vector
  - Newton iteration
  - Levenberg-Marquardt
  - Sparse Levenberg-Marquardt

NEWTON ITERATION

Taylor approximation:

The function is locally linear

e_0 is $f(P_0)$ - X

variables can be weighted by the covariance matrix $\boldsymbol{\Sigma}$

Jacobian

$$f(\mathrm{P}_0 + \Delta) \approx f(\mathrm{P}_0) + \mathrm{J}\Delta \qquad \mathrm{J} = \frac{\partial \mathrm{X}}{\partial \mathrm{P}}$$

$$\left\| \mathrm{X} - f(\mathrm{P}_1) \right\|$$

$$\left\| \mathrm{X} - f(\mathrm{P}_1) \right\| \approx \left\| \mathrm{X} - f(\mathrm{P}_0) - \mathrm{J}\Delta \right\| = \left\| e_0 - \mathrm{J}\Delta \right\|$$

$$\Rightarrow \mathrm{J}^{\mathrm{T}}\mathrm{J}\Delta = \mathrm{J}^{\mathrm{T}} e_0 \Rightarrow \Delta = \left(\mathrm{J}^{\mathrm{T}}\mathrm{J}\right)^{-1} \mathrm{J}^{\mathrm{T}} e_0$$

$$\boxed{\mathrm{P}_{i+1} = \mathrm{P}_i + \Delta} \qquad \boxed{\Delta = \left(\mathrm{J}^{\mathrm{T}}\mathrm{J}\right)^{-1} \mathrm{J}^{\mathrm{T}} e_0}$$

normal eq.

$$\Delta = \left(\mathrm{J}^{\mathrm{T}}\Sigma^{-1}\mathrm{J}\right)^{-1} \mathrm{J}^{\mathrm{T}}\Sigma^{-1} e_0$$

NEWTON ITERATION

- $P_{i+1} = P_i + \Delta_i$
- with Δi the solution of the normal equation evaluated at Pi
- Assuming it has a well defined minimum value, it will converge to the minimum, unfortunately, typically to the local minimum
- Very dependent of the initialization P_0

LEVENBERG-MARQUARDT

Normal equations

$$J^T J \Delta = N \Delta = J^T e_0$$

Augmented normal equations

$$N' \Delta = J^T e_0 \qquad N' = J^T J + \lambda \operatorname{diag}(J^T J)$$

$$\lambda_0 = 10^{-3}$$

$\text{success}: \lambda_{i+1} = \lambda_i / 10$ accept

$\text{failure}: \lambda_i = 10\lambda_i$ solve again

λ small ~ Newton (quadratic convergence)

λ large ~ descent (guaranteed decrease)

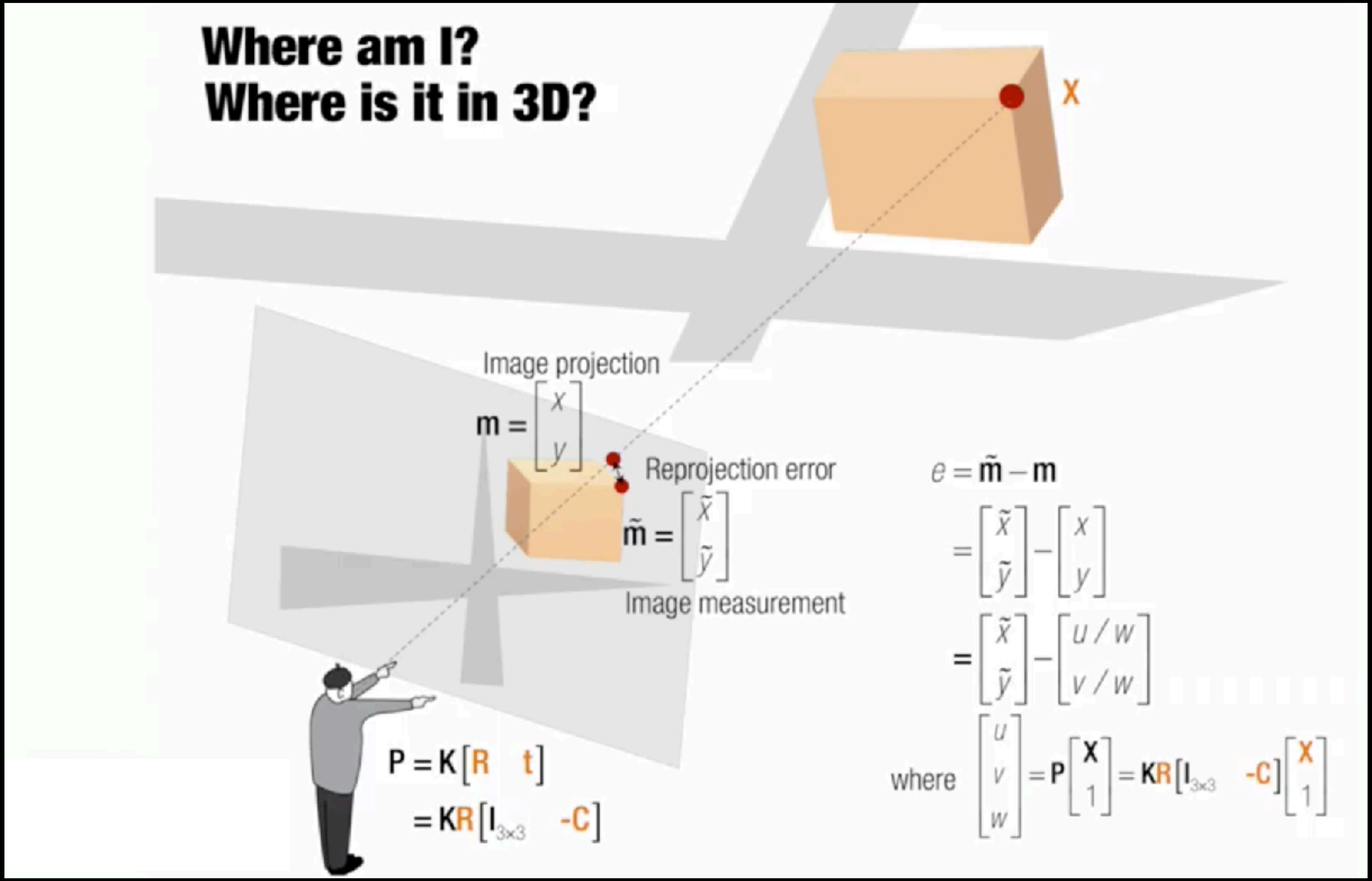
LEVENBERG-MARQUARDT

Requirements for minimization

- Function to compute f
- Start value P_0
- Optionally, function to compute J

  (but numerical ok, too)

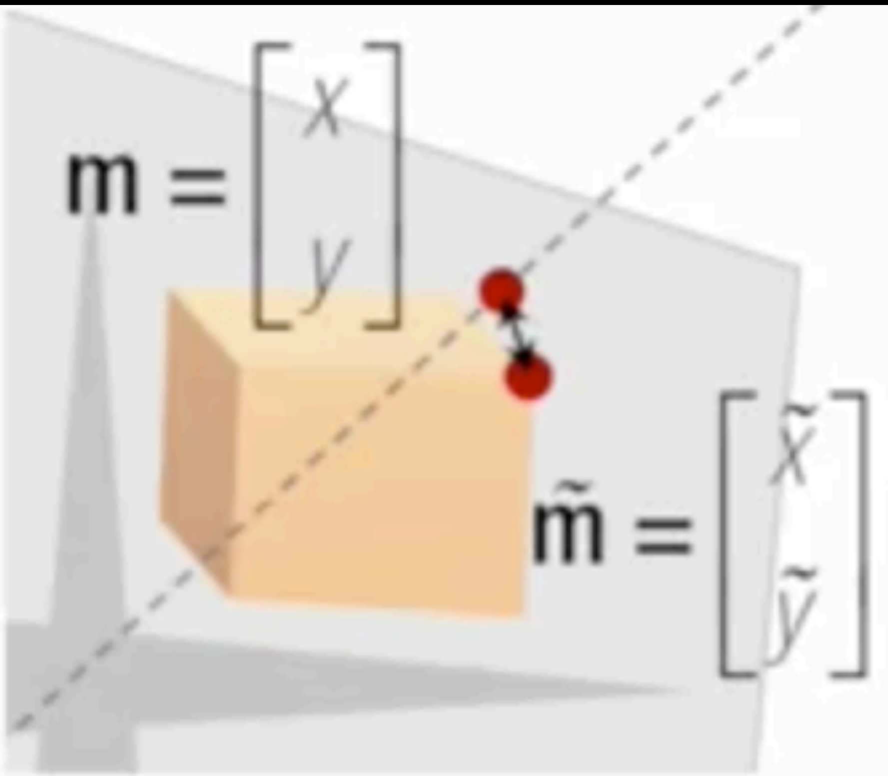
TEXT


Where am I?
Where is it in 3D?
X
Image projection
m = [x y]
Reprojection error
m̃ = [x̃ ỹ]
Image measurement
e = m̃ − m
= [x̃ ỹ] − [x y]
= [x̃ ỹ] − [u/w v/w]
where [u v w] = P[X 1] = KR[I 3×3 -C][X 1]
P = K[R t]
= KR[I 3×3 -C]

TEXT

Reprojection error

$$e = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u/w \\ v/w \end{bmatrix} \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{P}\begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{R}\begin{bmatrix} \mathbf{I}_{3\times3} & -\mathbf{C} \end{bmatrix}\begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

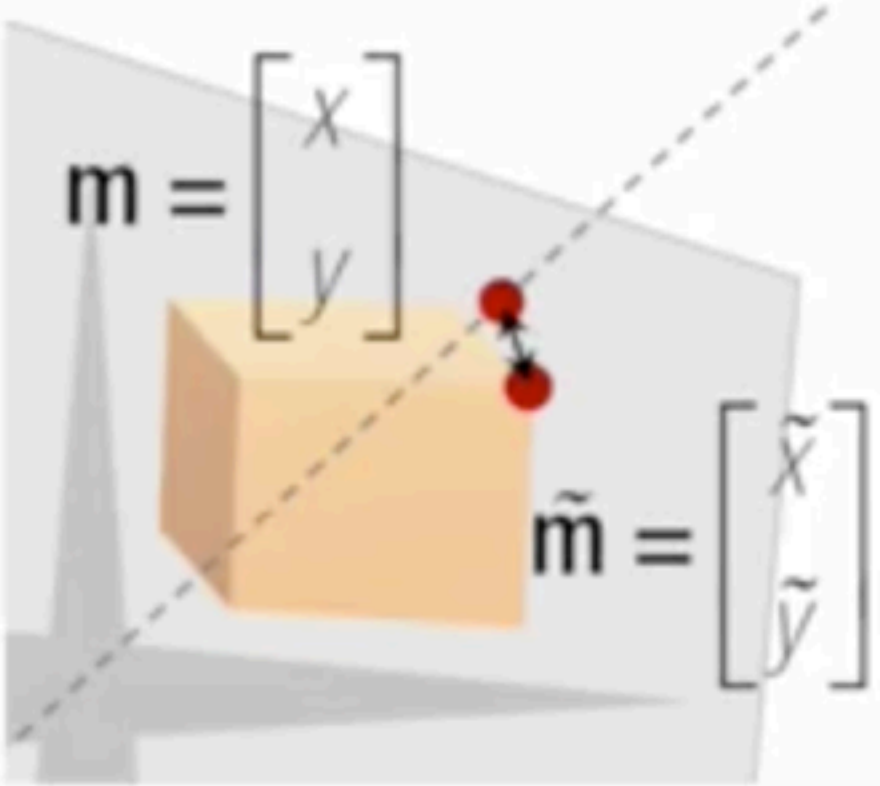


$$\underset{\mathbf{R},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u(\mathbf{R},\mathbf{C},\mathbf{X})/w(\mathbf{R},\mathbf{C},\mathbf{X}) \\ v(\mathbf{R},\mathbf{C},\mathbf{X})/w(\mathbf{R},\mathbf{C},\mathbf{X}) \end{bmatrix} \right\|^2$$

$$= \underset{\mathbf{q},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})/w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \\ v(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})/w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \end{bmatrix} \right\|^2 \quad \text{: Quaternion parameterization}$$

Reprojection error

$$e = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u/w \\ v/w \end{bmatrix} \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{P} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{R} \begin{bmatrix} \mathbf{I}_{3\times 3} & -\mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$



$$\underset{\mathbf{R},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u(\mathbf{R},\mathbf{C},\mathbf{X}) / w(\mathbf{R},\mathbf{C},\mathbf{X}) \\ v(\mathbf{R},\mathbf{C},\mathbf{X}) / w(\mathbf{R},\mathbf{C},\mathbf{X}) \end{bmatrix} \right\|^2$$

$$= \underset{\mathbf{q},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \underbrace{\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix}}_{\mathbf{b}} - \underbrace{\begin{bmatrix} u(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) / w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \\ v(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) / w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \end{bmatrix}}_{\mathbf{f}(\mathbf{R},\mathbf{C},\mathbf{X})} \right\|^2 \quad \text{: Quaternion parameterization}$$

TEXT

Recall

Nonlinear least squares:

$$\underset{x}{\text{minimize}} \|f(x) - b\|^2 = \underset{x}{\text{minimize}}\, f(x)^T f(x) - 2b^T f(x)$$

$$\longrightarrow \left.\frac{\partial E}{\partial x}\right|_{x^*} = 2\frac{\partial f(x)}{\partial x}^T f(x) - 2\frac{\partial f(x)}{\partial x}^T b = 0$$

Taylor expansion around x:

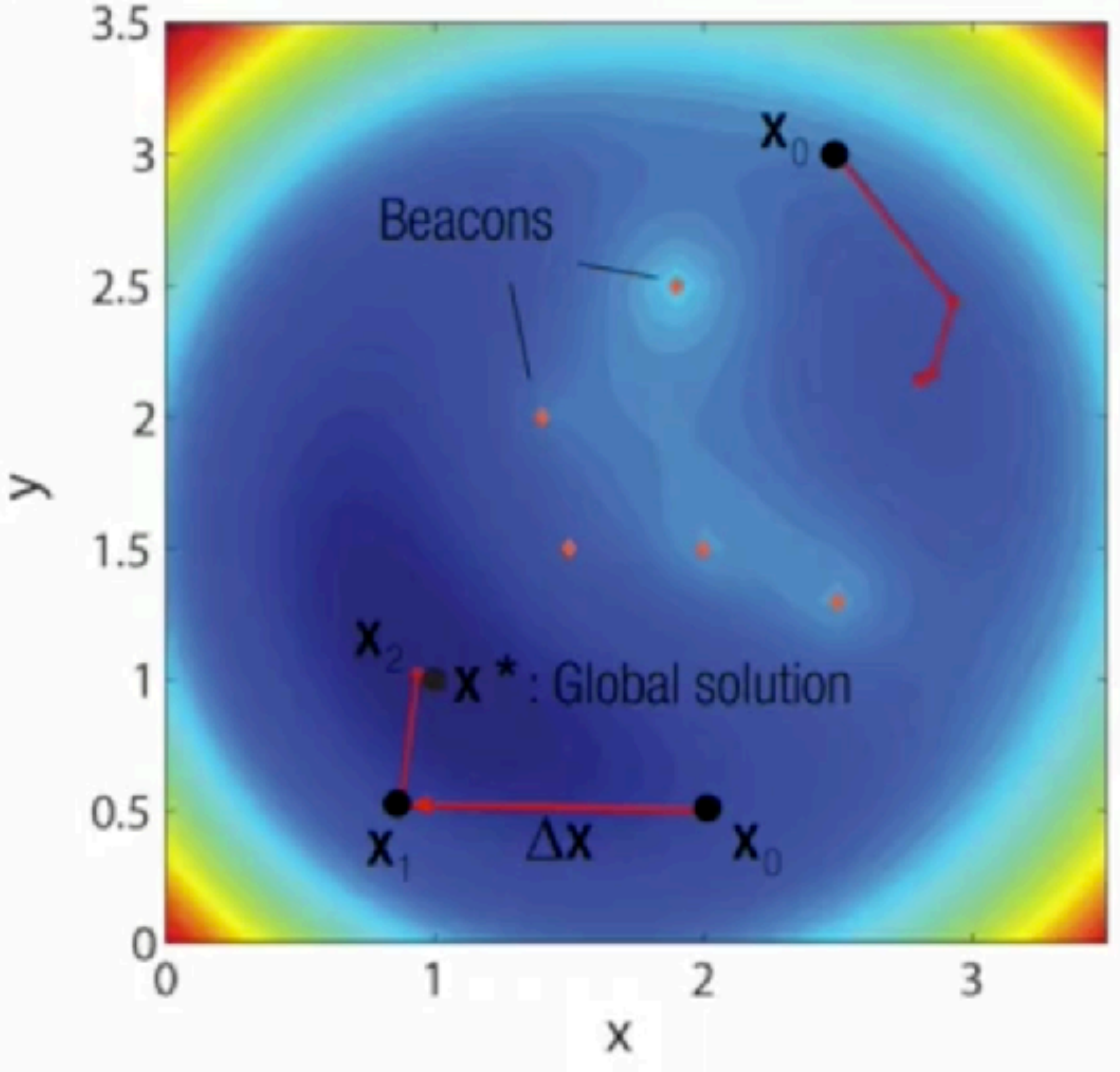
$$f(x + \Delta x) \approx f(x) + \frac{\partial f(x)}{\partial x}\Delta x$$

where $\boxed{\Delta x = \left(J^T J\right)^{-1} J^T (b - f(x))}$

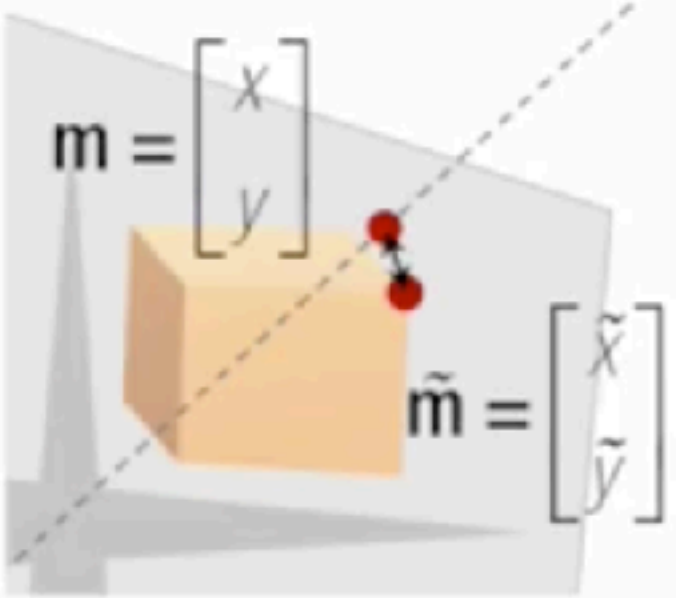
Normal equation

$$J = \frac{\partial f(x)}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$: Jacobian matrix

Example:
Localization using range data from beacons



Reprojection error



$$e = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u/w \\ v/w \end{bmatrix} \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{P}\begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{KR}\begin{bmatrix} \mathbf{I}_{3\times 3} & -\mathbf{C} \end{bmatrix}\begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

$$\underset{\mathbf{q},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) / w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \\ v(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) / w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \end{bmatrix} \right\|^2 = \underset{\mathbf{q},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \mathbf{b} - \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \right\|^2$$

$$\mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) = \begin{bmatrix} u/w \\ u/w \end{bmatrix} \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}[\mathbf{X}-\mathbf{C}]$$

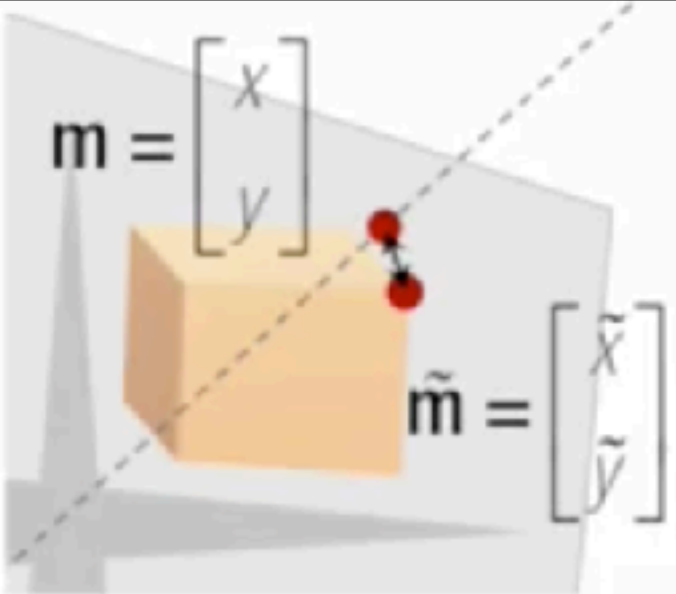
$$= \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} [\mathbf{X}-\mathbf{C}]$$

$$= \begin{bmatrix} fr_{11} + p_x r_{31} & fr_{12} + p_x r_{32} & fr_{13} + p_x r_{33} \\ fr_{21} + p_y r_{31} & fr_{22} + p_y r_{32} & fr_{23} + p_y r_{33} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} [\mathbf{X}-\mathbf{C}]$$

TEXT

Reprojection error

$$e = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u/w \\ v/w \end{bmatrix} \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{P}\begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{KR}\begin{bmatrix} \mathbf{I}_{3\times 3} & -\mathbf{C} \end{bmatrix}\begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$



$$\underset{\mathbf{q},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) / w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \\ v(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) / w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \end{bmatrix} \right\|^2 = \underset{\mathbf{q},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \mathbf{b} - \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \right\|^2$$

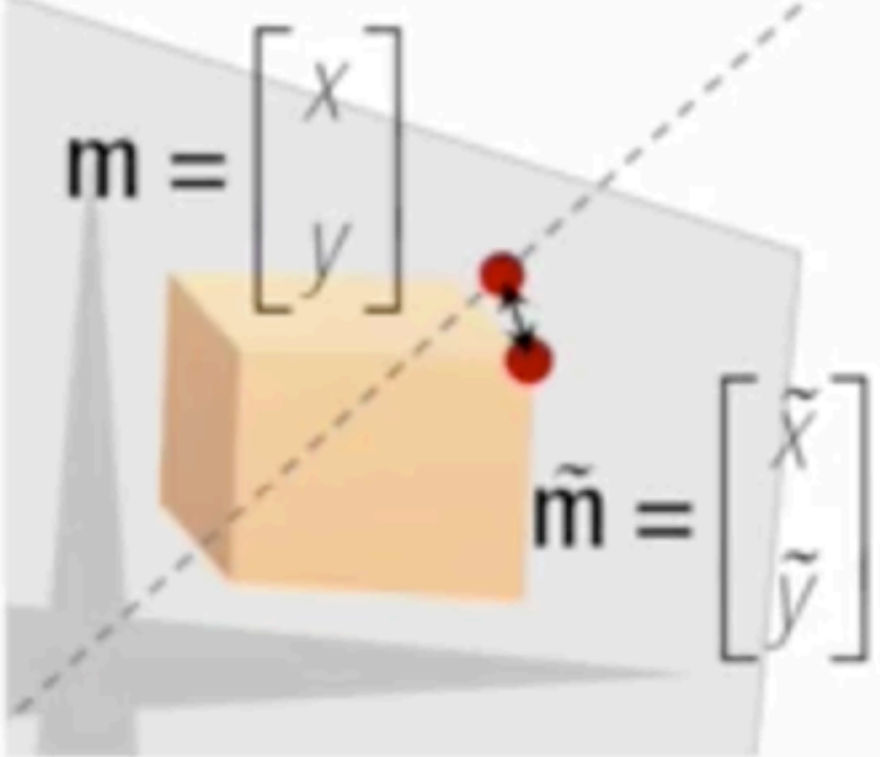
$$\mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) = \begin{bmatrix} u/w \\ u/w \end{bmatrix}$$

$$\text{where} \quad \begin{aligned} u &= \begin{bmatrix} fr_{11} + p_x r_{31} & fr_{12} + p_x r_{32} & fr_{13} + p_x r_{33} \end{bmatrix}\begin{bmatrix} \mathbf{X} - \mathbf{C} \end{bmatrix} \\ v &= \begin{bmatrix} fr_{21} + p_y r_{31} & fr_{22} + p_y r_{32} & fr_{23} + p_y r_{33} \end{bmatrix}\begin{bmatrix} \mathbf{X} - \mathbf{C} \end{bmatrix} \\ w &= \begin{bmatrix} r_{31} & r_{32} & r_{33} \end{bmatrix}\begin{bmatrix} \mathbf{X} - \mathbf{C} \end{bmatrix} \end{aligned}$$

TEXT

Reprojection error

$$e = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u/w \\ v/w \end{bmatrix} \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{P} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{R}\begin{bmatrix} \mathbf{I}_{3\times 3} & -\mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$



$$\underset{\mathbf{q},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} - \begin{bmatrix} u(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) / w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \\ v(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) / w(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \end{bmatrix} \right\|^2 = \underset{\mathbf{q},\mathbf{C},\mathbf{X}}{\text{minimize}} \left\| \mathbf{b} - \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X}) \right\|^2$$

$$\mathbf{J} = \begin{bmatrix} \dfrac{\partial \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{q}} & \dfrac{\partial \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{C}} & \dfrac{\partial \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{X}} \end{bmatrix}$$

$$= \begin{bmatrix} \underbrace{\dfrac{\partial \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{R}}}_{2\text{x}9} \underbrace{\dfrac{\partial \mathbf{R}}{\partial \mathbf{q}}}_{9\text{x}4} & \underbrace{\dfrac{\partial \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{C}}}_{2\text{x}3} & \underbrace{\dfrac{\partial \mathbf{f}(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{X}}}_{2\text{x}3} \end{bmatrix}$$

$$J = \left[\underbrace{\frac{\partial f(R(q),C,X)}{\partial R}}_{2\times 9} \underbrace{\frac{\partial R}{\partial q}}_{9\times 4} \quad \underbrace{\frac{\partial f(R(q),C,X)}{\partial C}}_{2\times 3} \quad \underbrace{\frac{\partial f(R(q),C,X)}{\partial X}}_{2\times 3} \right]$$

$$f(R(q),C,X) = \begin{bmatrix} u/w \\ u/w \end{bmatrix} \quad \text{where} \quad \begin{aligned} u &= \begin{bmatrix} fr_{11}+p_x r_{31} & fr_{12}+p_x r_{32} & fr_{13}+p_x r_{33} \end{bmatrix}[X-C] \\ v &= \begin{bmatrix} fr_{21}+p_y r_{31} & fr_{22}+p_y r_{32} & fr_{23}+p_y r_{33} \end{bmatrix}[X-C] \\ w &= \begin{bmatrix} r_{31} & r_{32} & r_{33} \end{bmatrix}[X-C] \end{aligned}$$

$$\underbrace{\frac{\partial f(R(q),C,X)}{\partial C}}_{2\times 3} = \begin{bmatrix} \dfrac{w\frac{\partial u}{\partial C} - u\frac{\partial w}{\partial C}}{w^2} \\ \dfrac{w\frac{\partial v}{\partial C} - v\frac{\partial w}{\partial C}}{w^2} \end{bmatrix} \quad \text{where} \quad \begin{aligned} \frac{\partial u}{\partial C} &= -\begin{bmatrix} fr_{11}+p_x r_{31} & fr_{12}+p_x r_{32} & fr_{13}+p_x r_{33} \end{bmatrix} \\ \frac{\partial v}{\partial C} &= -\begin{bmatrix} fr_{21}+p_y r_{31} & fr_{22}+p_y r_{32} & fr_{23}+p_y r_{33} \end{bmatrix} \\ \frac{\partial w}{\partial C} &= -\begin{bmatrix} r_{31} & r_{32} & r_{33} \end{bmatrix} \end{aligned}$$

TEXT

$$\mathbf{J} = \begin{bmatrix} \underbrace{\frac{\partial f(R(q),C,X)}{\partial R}}_{2\times 9} \underbrace{\frac{\partial R}{\partial q}}_{9\times 4} & \underbrace{\frac{\partial f(R(q),C,X)}{\partial C}}_{2\times 3} & \underbrace{\frac{\partial f(R(q),C,X)}{\partial X}}_{2\times 3} \end{bmatrix}$$

$$f(R(q),C,X) = \begin{bmatrix} u/w \\ u/w \end{bmatrix} \quad \text{where} \quad \begin{aligned} u &= \begin{bmatrix} fr_{11} + p_x r_{31} & fr_{12} + p_x r_{32} & fr_{13} + p_x r_{33} \end{bmatrix}\begin{bmatrix} \mathbf{X} - \mathbf{C} \end{bmatrix} \\ v &= \begin{bmatrix} fr_{21} + p_y r_{31} & fr_{22} + p_y r_{32} & fr_{23} + p_y r_{33} \end{bmatrix}\begin{bmatrix} \mathbf{X} - \mathbf{C} \end{bmatrix} \\ w &= \begin{bmatrix} r_{31} & r_{32} & r_{33} \end{bmatrix}\begin{bmatrix} \mathbf{X} - \mathbf{C} \end{bmatrix} \end{aligned}$$

$$\underbrace{\frac{\partial f(R(q),C,X)}{\partial \mathbf{X}}}_{2\times 3} = \begin{bmatrix} \dfrac{w\dfrac{\partial u}{\partial \mathbf{X}} - u\dfrac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \dfrac{w\dfrac{\partial v}{\partial \mathbf{X}} - v\dfrac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix} \quad \text{where} \quad \begin{aligned} \frac{\partial u}{\partial \mathbf{X}} &= \begin{bmatrix} fr_{11} + p_x r_{31} & fr_{12} + p_x r_{32} & fr_{13} + p_x r_{33} \end{bmatrix} \\ \frac{\partial v}{\partial \mathbf{X}} &= \begin{bmatrix} fr_{21} + p_y r_{31} & fr_{22} + p_y r_{32} & fr_{23} + p_y r_{33} \end{bmatrix} \\ \frac{\partial w}{\partial \mathbf{X}} &= \begin{bmatrix} r_{31} & r_{32} & r_{33} \end{bmatrix} \end{aligned}$$

TEXT

$$J = \left[\underbrace{\frac{\partial f(R(q), C, X)}{\partial R}}_{2x9} \underbrace{\frac{\partial R}{\partial q}}_{9x4} \quad \underbrace{\frac{\partial f(R(q), C, X)}{\partial C}}_{2x3} \quad \underbrace{\frac{\partial f(R(q), C, X)}{\partial X}}_{2x3} \right]$$

$$f(R(q), C, X) = \begin{bmatrix} u/w \\ u/w \end{bmatrix} \quad \text{where} \quad \begin{aligned} u &= \begin{bmatrix} fr_{11} + p_x r_{31} & fr_{12} + p_x r_{32} & fr_{13} + p_x r_{33} \end{bmatrix} [\mathbf{X} - \mathbf{C}] \\ v &= \begin{bmatrix} fr_{21} + p_y r_{31} & fr_{22} + p_y r_{32} & fr_{23} + p_y r_{33} \end{bmatrix} [\mathbf{X} - \mathbf{C}] \\ w &= \begin{bmatrix} r_{31} & r_{32} & r_{33} \end{bmatrix} [\mathbf{X} - \mathbf{C}] \end{aligned}$$

$$\underbrace{\frac{\partial f(R(q), C, X)}{\partial R}}_{2x9} = \begin{bmatrix} \dfrac{w \frac{\partial u}{\partial \mathbf{R}} - u \frac{\partial w}{\partial \mathbf{R}}}{w^2} \\ \dfrac{w \frac{\partial v}{\partial \mathbf{R}} - v \frac{\partial w}{\partial \mathbf{R}}}{w^2} \end{bmatrix} \quad \text{where} \quad \begin{aligned} \frac{\partial u}{\partial \mathbf{R}} &= \begin{bmatrix} f(X_1 - C_1) & \mathbf{0}_{1\times 3} & p_x(X_3 - C_3) \end{bmatrix} \\ \frac{\partial v}{\partial \mathbf{R}} &= \begin{bmatrix} \mathbf{0}_{1\times 3} & f(X_1 - C_1) & p_x(X_3 - C_3) \end{bmatrix} \\ \frac{\partial w}{\partial \mathbf{R}} &= \begin{bmatrix} \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} & (X_3 - C_3) \end{bmatrix} \end{aligned}$$

$$\mathbf{J} = \begin{bmatrix} \underbrace{\frac{\partial f(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{R}}}_{2\text{x}9} \underbrace{\frac{\partial \mathbf{R}}{\partial \mathbf{q}}}_{9\text{x}4} & \underbrace{\frac{\partial f(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{C}}}_{2\text{x}3} & \underbrace{\frac{\partial f(\mathbf{R}(\mathbf{q}),\mathbf{C},\mathbf{X})}{\partial \mathbf{X}}}_{2\text{x}3} \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} 1-2q_z^2-2q_y^2 & -2q_zq_w+2q_yq_x & 2q_yq_w+2q_zq_x \\ 2q_xq_y+2q_wq_z & 1-2q_z^2-2q_x^2 & 2q_zq_y-2q_xq_w \\ 2q_xq_z-2q_wq_y & 2q_yq_z+2q_wq_x & 1-2q_y^2-2q_x^2 \end{bmatrix}$$

where $\mathbf{q} = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix}^{\mathrm{T}}$

$$\underbrace{\frac{\partial \mathbf{R}}{\partial \mathbf{q}}}_{9\text{x}4} = \begin{bmatrix} \frac{\partial r_{11}}{\partial \mathbf{q}} \\ \frac{\partial r_{12}}{\partial \mathbf{q}} \\ \vdots \\ \frac{\partial r_{33}}{\partial \mathbf{q}} \end{bmatrix}$$

where

$$\frac{\partial \mathbf{R}_{11}}{\partial \mathbf{q}} = \begin{bmatrix} 0 & -4q_y & -4q_z & 0 \end{bmatrix} \qquad \frac{\partial \mathbf{R}_{23}}{\partial \mathbf{q}} = \begin{bmatrix} -2q_w & 2q_z & 2q_y & 2q_x \end{bmatrix}$$

$$\frac{\partial \mathbf{R}_{12}}{\partial \mathbf{q}} = \begin{bmatrix} 2q_y & 2q_x & -2q_w & -2q_z \end{bmatrix} \qquad \frac{\partial \mathbf{R}_{31}}{\partial \mathbf{q}} = \begin{bmatrix} 2q_z & -2q_w & 2q_x & -2q_y \end{bmatrix}$$

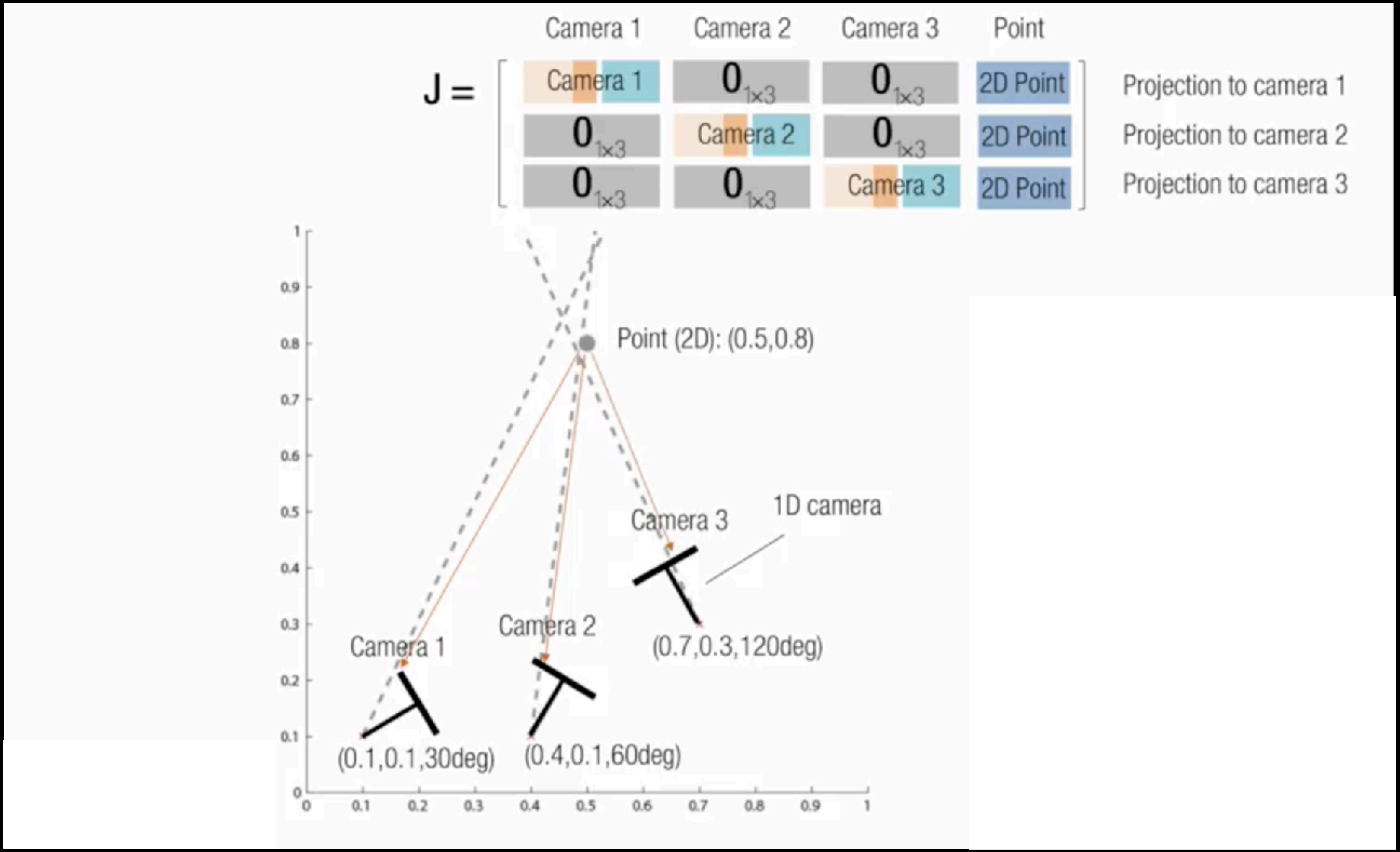
$$\frac{\partial \mathbf{R}_{13}}{\partial \mathbf{q}} = \begin{bmatrix} 2q_z & 2q_w & 2q_x & 2q_y \end{bmatrix} \qquad \frac{\partial \mathbf{R}_{33}}{\partial \mathbf{q}} = \begin{bmatrix} -4q_x & -4q_y & 0 & 0 \end{bmatrix}$$

$$\frac{\partial \mathbf{R}_{21}}{\partial \mathbf{q}} = \begin{bmatrix} 2q_y & 2q_x & 2q_w & 2q_z \end{bmatrix} \qquad \frac{\partial \mathbf{R}_{32}}{\partial \mathbf{q}} = \begin{bmatrix} 2q_w & 2q_z & 2q_y & 2q_x \end{bmatrix}$$

$$\frac{\partial \mathbf{R}_{22}}{\partial \mathbf{q}} = \begin{bmatrix} -4q_x & 0 & -4q_z & 0 \end{bmatrix}$$

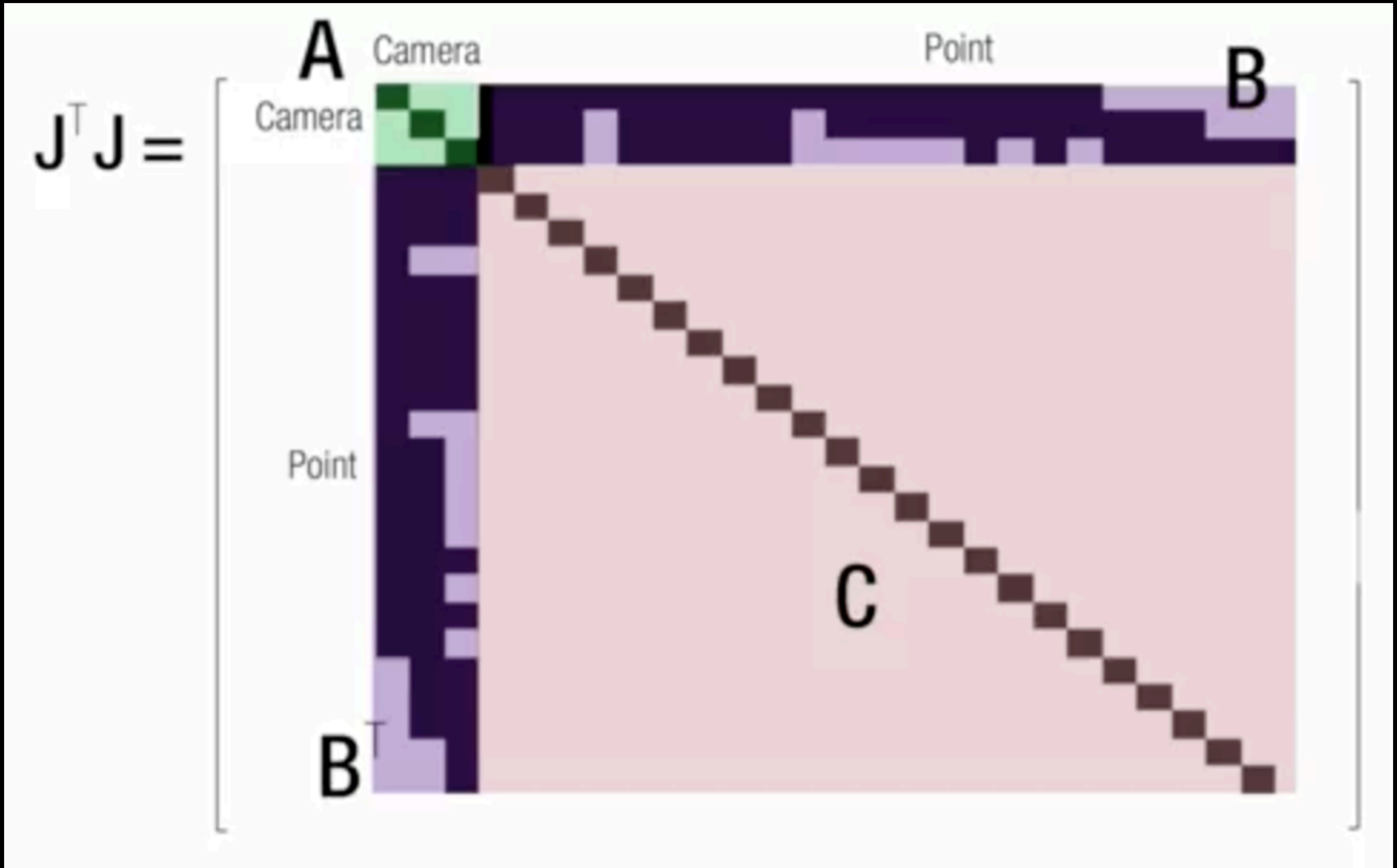
Penn Engineering

TEXT

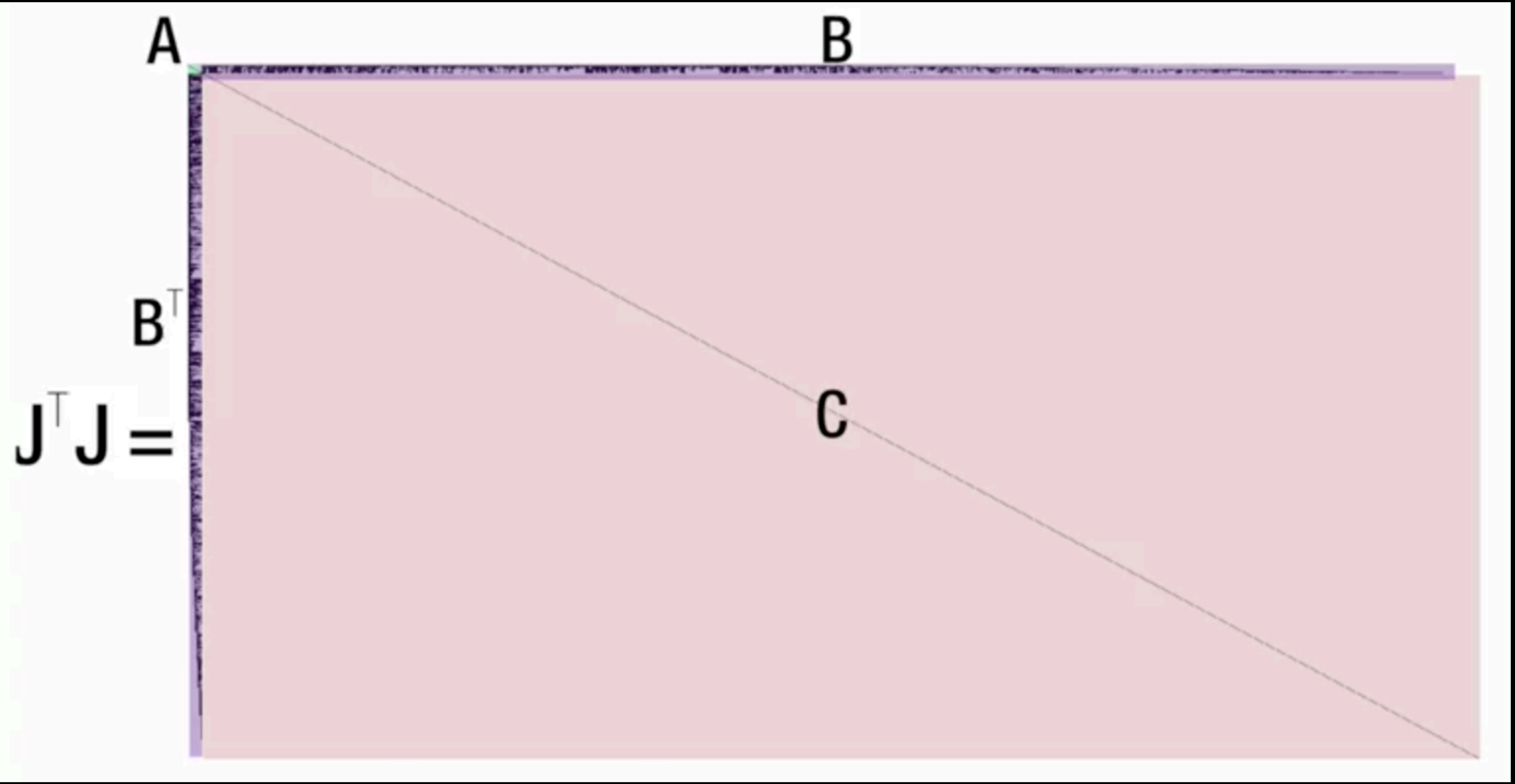


Camera 1
Camera 2
Camera 3
Point
J =
Camera 1
0 1x3
2D Point
Camera 2
Camera 3
Projection to camera 1
Projection to camera 2
Projection to camera 3
Point (2D): (0.5,0.8)
1D camera
(0.1,0.1,30deg)
(0.4,0.1,60deg)
(0.7,0.3,120deg)

TEXT



TEXT



SPARSE LEVENBERG-MARQUARDT

- N^3 complexity for solving $\Delta = N'^{-1} J^T e_0$
  - prohibitive for large problems

  (100 views 10,000 points ~30,000 unknowns)

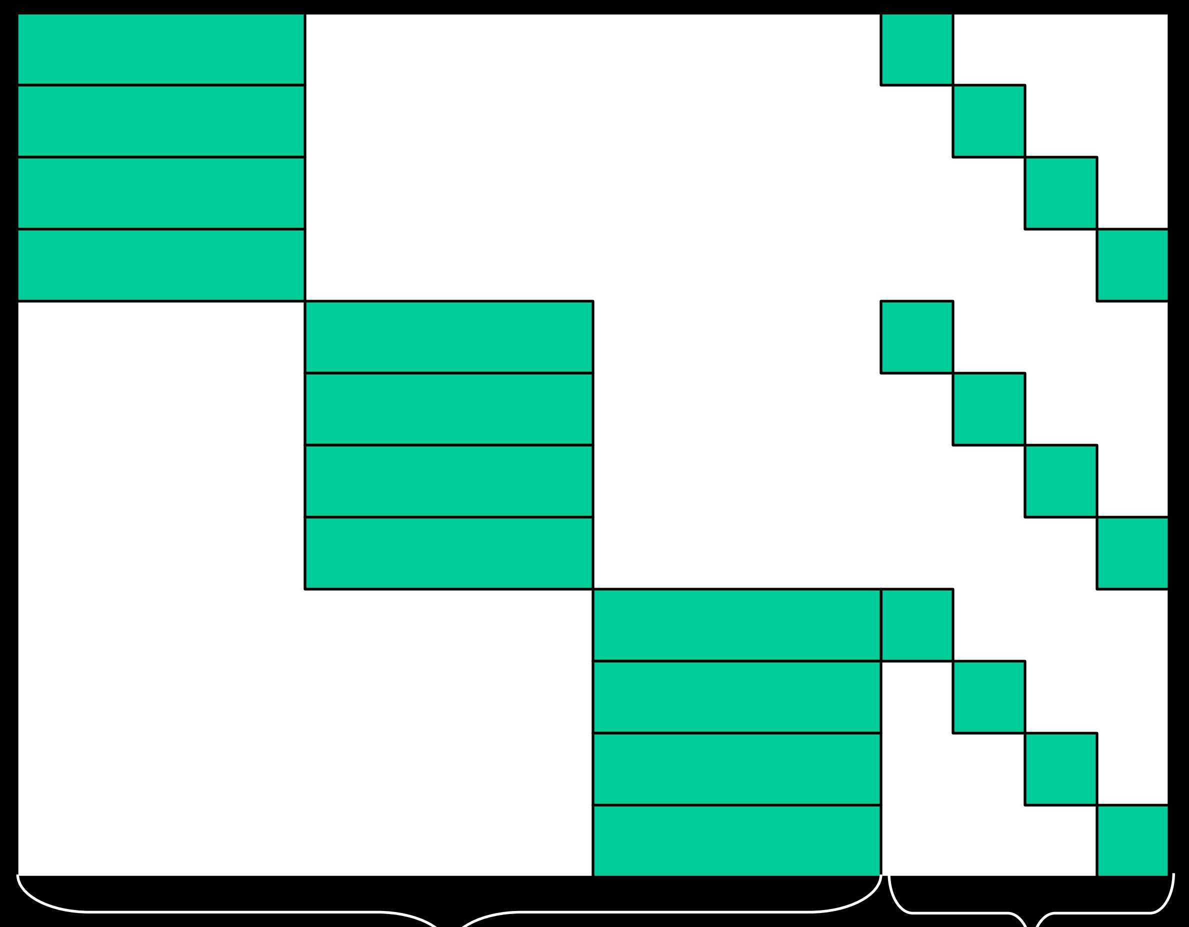
- Partition parameters
  - partition A
  - partition B (only dependent on A and itself)

SPARSE BUNDLE ADJUSTMENT

Jacobian of $\sum_{k=1}^{m}\sum_{i=1}^{n} D\left(m_{ki}, \hat{P}_k\left(\hat{M}_i\right)\right)^2$ has sparse block structure

im.pts.
view 1

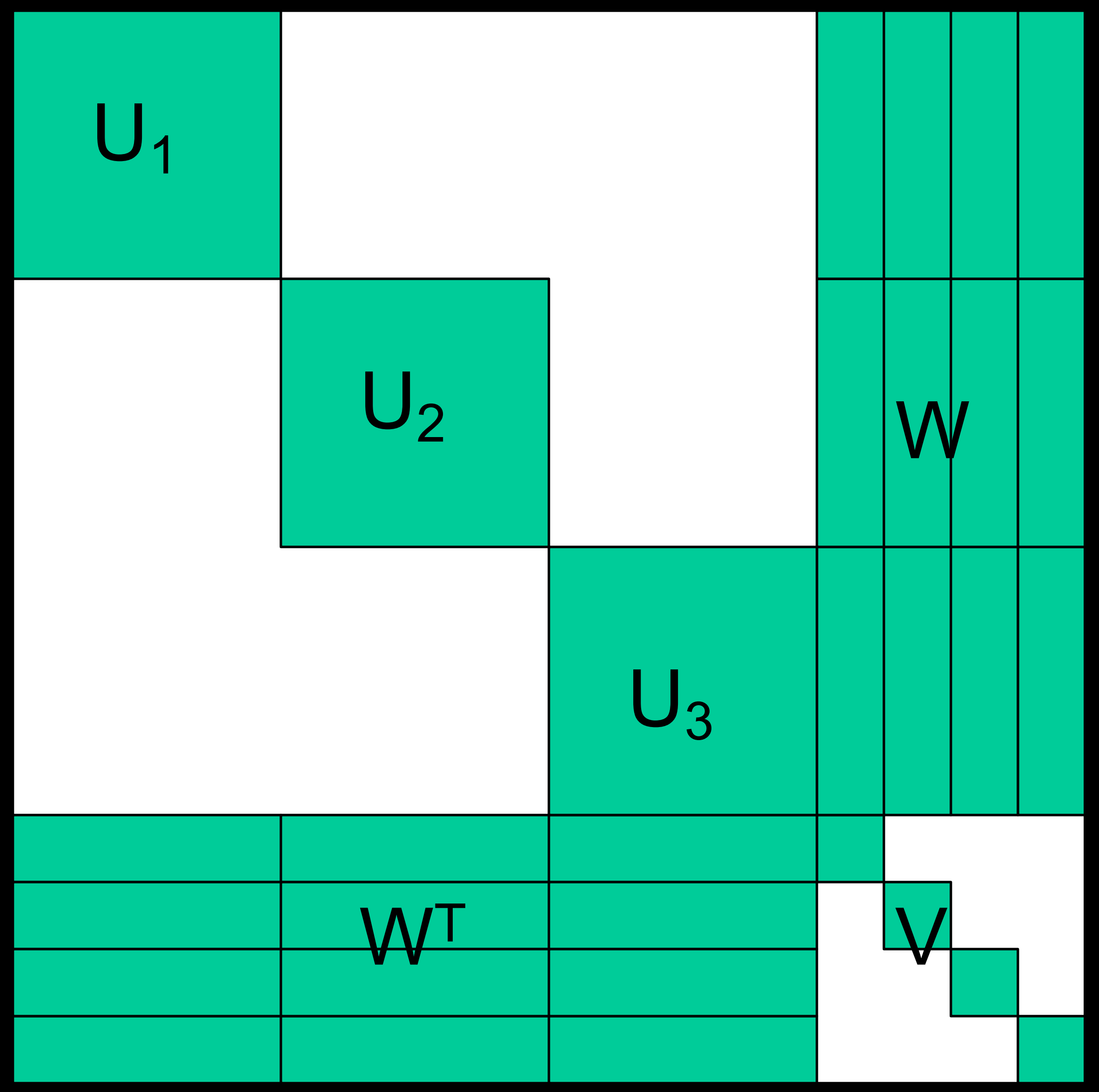
J =



12xm

3xn
(in general
much larger)

N = J^{T}J =



Needed for non-linear minimization

SPARSE BUNDLE ADJUSTMENT

residuals: $D(\mathtt{m}_{ki}, \hat{\mathbf{P}}_k \hat{\mathtt{M}}_i)^2$

normal equations: $\begin{bmatrix} \mathbf{U} & \mathbf{W} \\ \mathbf{W}^\top & \mathbf{V} \end{bmatrix} \begin{bmatrix} \Delta(\mathbf{P}) \\ \Delta(\mathtt{M}) \end{bmatrix} = \begin{bmatrix} \epsilon(\mathbf{P}) \\ \epsilon(\mathtt{M}) \end{bmatrix}$

with

$$\mathbf{U}_k = \sum_i \left(\frac{\partial \hat{\mathtt{m}}_{ki}}{\partial \hat{\mathbf{P}}_k}\right)^\top \frac{\partial \hat{\mathtt{m}}_{ki}}{\partial \hat{\mathbf{P}}_k}$$

$$\mathbf{V}_i = \sum_k \left(\frac{\partial \hat{\mathtt{m}}_{ki}}{\partial \hat{\mathtt{M}}_i}\right)^\top \frac{\partial \hat{\mathtt{m}}_{ki}}{\partial \hat{\mathtt{M}}_i}$$

$$\mathbf{W}_{ki} = \left(\frac{\partial \hat{\mathtt{m}}_{ki}}{\partial \hat{\mathbf{P}}_k}\right)^\top \frac{\partial \hat{\mathtt{m}}_{ki}}{\partial \hat{\mathtt{M}}_i}$$

$$\epsilon(\mathbf{P}_k) = \sum_i \left(\frac{\partial \hat{\mathtt{m}}_{ki}}{\partial \hat{\mathbf{P}}_k}\right)^\top \epsilon_{ki}$$

$$\epsilon(\mathtt{M}_i) = \sum_i \left(\frac{\partial \hat{\mathtt{m}}_{ki}}{\partial \hat{\mathtt{M}}_i}\right)^\top \epsilon_{ki}$$

SPARSE BUNDLE ADJUSTMENT

normal equations:

$$\begin{bmatrix} \mathbf{I} & -\mathbf{W}\mathbf{V}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U} & \mathbf{W} \\ \mathbf{W}^\top & \mathbf{V} \end{bmatrix} \begin{bmatrix} \Delta(\mathbf{P}) \\ \Delta(\mathtt{M}) \end{bmatrix} = \begin{bmatrix} \epsilon(\mathbf{P}) \\ \epsilon(\mathtt{M}) \end{bmatrix}$$

modified normal equations:

$$\begin{bmatrix} \mathbf{U} - \mathbf{W}\mathbf{V}^{-1}\mathbf{W}^\top & \mathbf{0} \\ \mathbf{W}^\top & \mathbf{V} \end{bmatrix} \begin{bmatrix} \Delta(\mathbf{P}) \\ \Delta(\mathtt{M}) \end{bmatrix} = \begin{bmatrix} \epsilon(\mathbf{P}) - \mathbf{W}\mathbf{V}^{-1}\epsilon(\mathtt{M}) \\ \epsilon(\mathtt{M}) \end{bmatrix}$$

solve in two parts:

$$\left(\mathbf{U} - \mathbf{W}\mathbf{V}^{-1}\mathbf{W}^\top\right)\Delta(\mathbf{P}) = \epsilon(\mathbf{P}) - \mathbf{W}\mathbf{V}^{-1}\epsilon(\mathtt{M})$$

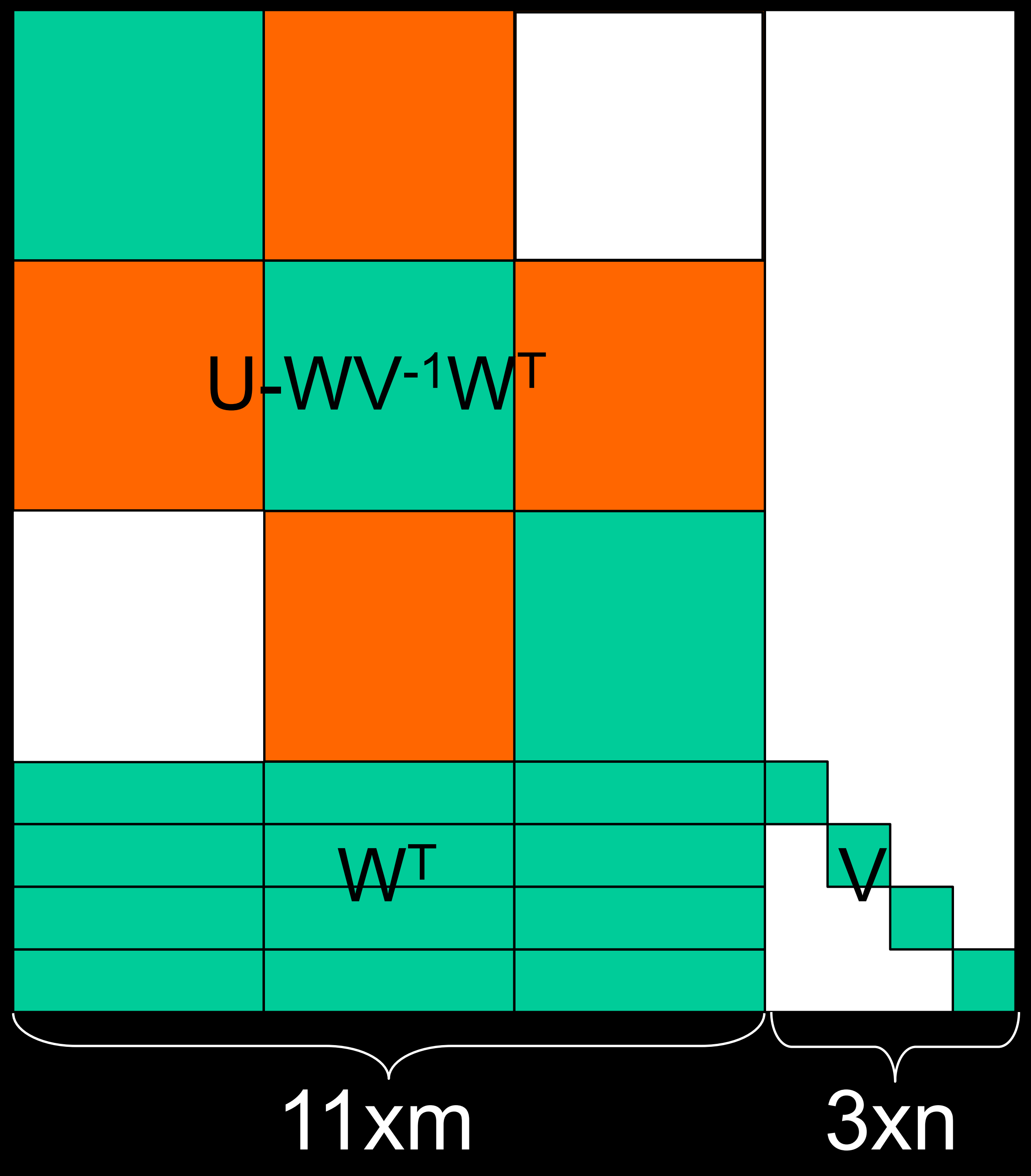
$$\Delta(\mathtt{M}) = \mathbf{V}^{-1}\left(\epsilon(\mathtt{M}) - \mathbf{W}^\top\Delta(\mathbf{P})\right)$$

SPARSE BUNDLE ADJUSTMENT

- **Eliminate dependence of camera/motion parameters on structure parameters**

Note in general 3n >> 11m

$$\begin{bmatrix} \mathbf{I} & -\mathbf{W}\mathbf{V}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \times \mathbf{N} =$$



Allows much more efficient computations

e.g. 100 views,10000 points,

solve ±1000x1000, not ±30000x30000

Often still band diagonal

use sparse linear algebra algorithms

SPARSE BUNDLE ADJUSTMENT

normal equations:

$$\begin{bmatrix} \mathbf{I} & -\mathbf{W}\mathbf{V}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U} & \mathbf{W} \\ \mathbf{W}^\top & \mathbf{V} \end{bmatrix} \begin{bmatrix} \Delta(\mathbf{P}) \\ \Delta(\mathtt{M}) \end{bmatrix} = \begin{bmatrix} \epsilon(\mathbf{P}) \\ \epsilon(\mathtt{M}) \end{bmatrix}$$

modified normal equations:

$$\begin{bmatrix} \mathbf{U} - \mathbf{W}\mathbf{V}^{-1}\mathbf{W}^\top & \mathbf{0} \\ \mathbf{W}^\top & \mathbf{V} \end{bmatrix} \begin{bmatrix} \Delta(\mathbf{P}) \\ \Delta(\mathtt{M}) \end{bmatrix} = \begin{bmatrix} \epsilon(\mathbf{P}) - \mathbf{W}\mathbf{V}^{-1}\epsilon(\mathtt{M}) \\ \epsilon(\mathtt{M}) \end{bmatrix}$$

solve in two parts:

$$\left(\mathbf{U} - \mathbf{W}\mathbf{V}^{-1}\mathbf{W}^\top\right)\Delta(\mathbf{P}) = \epsilon(\mathbf{P}) - \mathbf{W}\mathbf{V}^{-1}\epsilon(\mathtt{M})$$

$$\Delta(\mathtt{M}) = \mathbf{V}^{-1}\left(\epsilon(\mathbf{M}) - \mathbf{W}^\top\Delta(\mathbf{P})\right)$$

DETAILS

7 POINTS ALGORITHM FOR F

- In fact the two view structure (or the fundamental matrix) only has seven degrees of freedom. If one is prepared to solve non-linear equations, seven points must thus be sufficient to solve for it. In this case the rank-2 constraint must be enforced during the computations. A similar approach as in the previous section can be followed to characterize the right null-space of the system of linear equations originating from the seven point correspondences. This space can be parameterized as follows $v1+ \lambda v1$ or $F1+ \lambda F2$ with $v1$ and $v2$ being the two last columns of V (obtained through SVD) and F1 respectively F2 the corresponding matrices. The rank-2 constraint is then written as which is a polynomial of degree 3 in λ.

- This can simply be solved analytically. There are always 1 or 3 real solutions. The special case F1 (which is not covered by this parameterization) is easily checked separately, i.e. it should have rank-2. If more than one solution is obtained then more points are needed to obtain the true fundamental matrix.

THE MINIMUM CASE – 7 POINT CORRESPONDENCES

$$\begin{bmatrix} x'_1 x_1 & x'_1 y_1 & x'_1 & y'_1 x_1 & y'_1 y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_7 x_7 & x'_7 y_7 & x'_7 & y'_7 x_7 & y'_7 y_7 & y'_7 & x_7 & y_7 & 1 \end{bmatrix} \mathbf{f} = 0$$

$$A = U_{7x7}\operatorname{diag}(\sigma_1,\ldots,\sigma_7,0,0)V_{9x9}^{T}$$

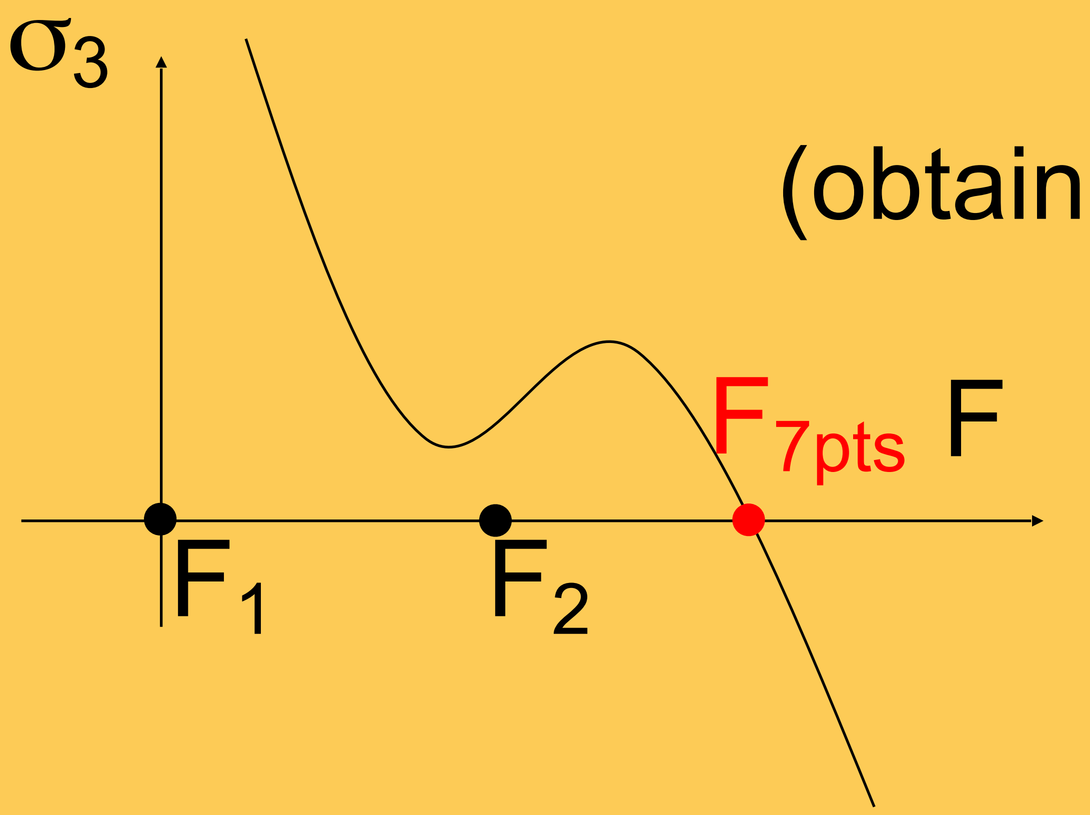
$$\Rightarrow A[V_8 V_9] = 0_{9x2} \qquad (\text{e.g.}\, V^T V_8 = [000000010]^T)$$

$$x_i^T (F_1 + \lambda F_2) x_i = 0, \forall i = 1 \ldots 7$$

one parameter family of solutions

but $F_1+\lambda F_2$ not automatically rank 2

THE MINIMUM CASE – IMPOSE RANK 2



(obtain 1 or 3 solutions)

$$\det(F_1 + \lambda F_2) = a_3\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0$$ (cubic equation)

$$\det(F_1 + \lambda F_2) = \det F_2 \det(F_2^{-1}F_1 + \lambda I) = 0$$

Compute possible λ as eigenvalues of $F_2^{-1}F_1$
(only real solutions are potential solutions)

NON-LINEAR OPTIMISATION OF F WITH ALL POINTS

- Even then the error that is minimized is an algebraic error which has nor real "physical" meaning. It is always better to minimize a geometrically meaningful criterion. The error measure that immediately comes to mind is the distance between the points and the epipolar lines. Assuming that the noise on every feature point is independent zero-mean Gaussian with the same sigma for all points, the minimization of the following criterion yields a maximum likelihood solution:

$$\mathcal{C}(F) = \sum \left(D(\mathrm{m}', \mathbf{F}\mathrm{m})^2 + D(\mathrm{m}, \mathbf{F}^\top \mathrm{m}')^2 \right)$$

- with D(m,l') the orthogonal distance between the point m and the line l'. This criterion can be minimized through a Levenberg-Marquard algorithm. The results obtained through linear least-squares can be used for initialization.

THE DEGENERATE CASE FOR F

- The computation of the two-view geometry requires that the matches originate from a 3D scene and that the motion is more than a pure rotation. If the observed scene is planar, the fundamental matrix is only determined up to three degrees of freedom. The same is true when the camera motion is a pure rotation.
- In this last case -only having one center of projection- depth can not be observed. In the absence of noise the detection of these degenerate cases would not be too hard. Starting from real -and thus noisy- data, the problem is much harder since the remaining degrees of freedom in the equations are then determined by noise.

DOMINANT PLANE

DEALING WITH THE DOMINANT PLANES

- Take pictures without pure rotation
- Planar scenes are common, Scene related problems occur when (part of) the scene is purely planar. In this case it is not possible anymore to determine the epipolar geometry uniquely. If the scene is planar, the image motion can be fully described by a homography
- Since , $\mathbf{F} = [\mathbf{e}']_\times \mathbf{H}$ there is a 2 parameter family of solutions for the epipolar geometry. In practice robust techniques would pick a random solution based on the inclusion of some outliers.

DEALING WITH THE DOMINANT PLANES

- Detect where only planar features are being matched.
- The Geometric Robust Information Criterion (GRIC) model selection approach. The GRIC selects the model with the lowest score.
- The score of a model is obtained by summing two contributions.
- The first one is related to the goodness of the fit and the second one is related to the parsimony of the model.
- GRIC takes into account the number of inliers plus outliers n, the residuals e, the standard deviation of the measurement error , the dimension of the data r, the k number of motion model parameters and the dimension d of the structure

DEALING WITH THE DOMINANT PLANES

$$\mathrm{GRIC} = \sum \rho(e_i^2) + (nd\ln(r) + k\ln(rn)) \quad .$$

$$\rho(e^2) = \min\left(\frac{e^2}{\sigma^2}, 2(r-d)\right) \quad .$$

- *nd* ln (r) represents the penalty term for the structure having n times d parameters each estimated from r observations and k ln (rn) represents the penalty term for the motion model having k parameters estimated from rn observations.
- For each image pair GRIC(F) and GRIC(H) can be compared. If GRIC(H) yields the lowest value it is assumed that most matched features are located on a dominant plane and that a homography model is therefore appropriate. On the contrary, when GRIC(F) yields the lowest value one could assume that standard projective structure and motion recovery could be continued.

DEALING WITH THE DOMINANT PLANES

- They propose to use the GRIC criterion on triplets of views (r=6). On the one hand we have GRIC(PPP) based on a model containing 3 projection matrices (up to a projective ambiguity) with $k=3 \times 11 - 15 = 18$ and $d=3$ (note that using a model based on the trifocal tensor would be equivalent), on the other hand we have GRIC(HH) based on a model containing 2 homographies with $k=2 \times 8= 16$ and $d=2$.

- To efficiently compute the MLE of both PPP and HH the sparse structure of the problem is exploited (similar to bundle adjustment). We can now differentiate between two different cases:

- Case A: GRIC(PPP) < GRIC(HH): three views observe general 3D structure.

- Case B: GRIC(PPP) > GRIC(HH): common structure between three views is planar.

PARTIAL PROJECTIVE STRUCTURE AND MOTION RECOVERY

```
case    AABAABBBBAAA
3D     PPPP        PPPPP
2D       HH HHHHHH
3D       PPPP
        FFFFFHHHHFFFF
```

- The sequence is first traversed and separated in subsequences. For subsequences with sufficient 3D structure (case A) projective structure and motion is recovered.
- When a triplet corresponds to case B, only planar features are tracked and reconstructed (in 2D).
- Note that the triplet 3-4-5 would cause an approach based on this to fail.

PARTIAL PROJECTIVE STRUCTURE AND MOTION RECOVERY

- Suppose the plane is labeled as a dominant plane from view based on features tracked in views (i-1,i, i+1). In general, some feature points $M_{\prod}$ located on $\prod$ will have been reconstructed in 3D from previous views (e.g. i and i-1). Therefore, the coefficients of $\prod$ can be computed from $M_{\prod}T \prod$= **0**

- Define $\mathbf{M}_{\prod}$ as the right null space $\prod$ of (4x3 matrix). $M_{\prod}$ represents 3 supporting points for the plane $\prod$ and let $m_{\prod}i$ = Pi $\mathbf{M}_{\prod}$ be the corresponding image projections. Define the homography $Hi_{\prod} = m_{\prod}i^{-1}$, then the 3D reconstruction of image points located in the plane $\prod$ are obtained as follows

$$\mathtt{M}_i = \mathbf{M}_{\Pi}\mathbf{H}_{i\Pi}\mathtt{m}_i$$

- Similarly, a feature mj seen in view j (> i) can be reconstructed as:

$$\mathtt{M}_j = \mathbf{M}_{\Pi}\mathbf{H}_{i\Pi}(\mathbf{H}^{\Pi}_{ij})^{-1}\mathtt{m}_j$$

- where

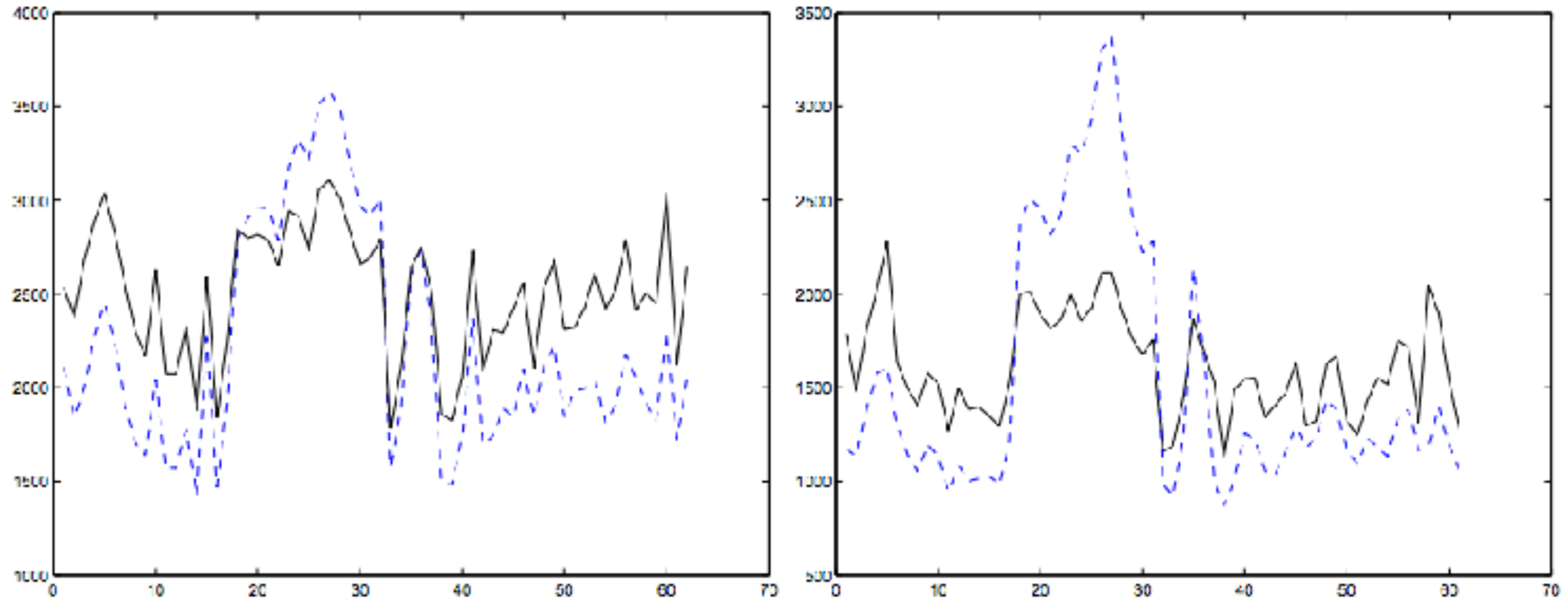
$$\mathbf{H}^{\Pi}_{ij} = \mathbf{H}^{\Pi}_{i(i+1)} \cdots \mathbf{H}^{\Pi}_{(j-1)j}.$$

COMBINED METRIC STRUCTURE AND MOTION RECOVERY

- Using the coupled self-calibration algorithm described in Section 6.3.1 it is possible to recover the metric structure of the different subsequences.
- Once the metric structure of the subsequences has been recovered, the pose of the camera can also be determined for the viewpoints observing only planar points. Since the intrinsics have been computed, a standard pose estimation algorithm can be used. We use Grunert's algorithm as described in [44]. To deal with outliers a robust approach was implemented [35].
- Finally, it becomes possible to align the structure and motion recovered for the separate subsequences based on common points. Note that these points are all located in a plane and therefore some precautions have to be taken to obtain results using linear equations. However, since 3 points form a basis in a metric 3D space, additional points out of the plane can easily be generated (i.e. using the vector product) and used to compute the relative transform using linear equations. Here again a robust approach is used.

EXAMPLE

Figure 5.7: Some of the 64 images of the *corner* sequence.



igure 5.8: Left: GRIC(**F**) (solid/black line) and GRIC(**H**) (dashed/blue line). Right: GRIC(**PPP**) olid/black line) and GRIC(**HH**) (dashed/blue line).

- GRIC values are given for and as well as for and . It can clearly be seen that –besides dealing with additional ambiguities– the triplet based analysis in general provides more discriminant results. It is also interesting to note that triplet 34-35-36 is clearly indicated as containing sufficiently general structure for the triplet-based approach while the pair-based approach marginally prefers to use the plane based model.

EXAMPLE

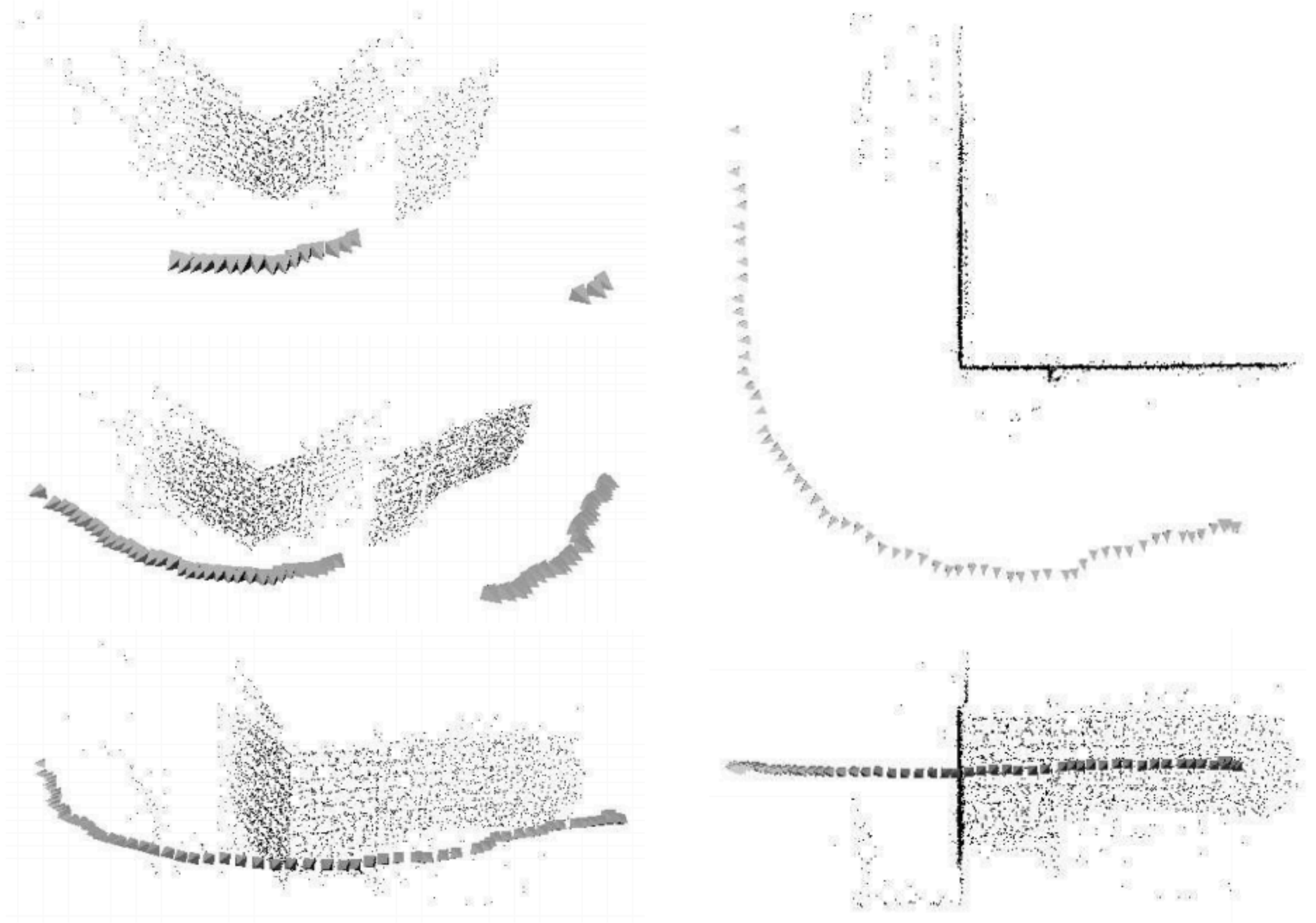


Figure 5.9: Left: different stages of the structure and motion recovery, Right: orthogonal views of the final result.

EXAMPLE

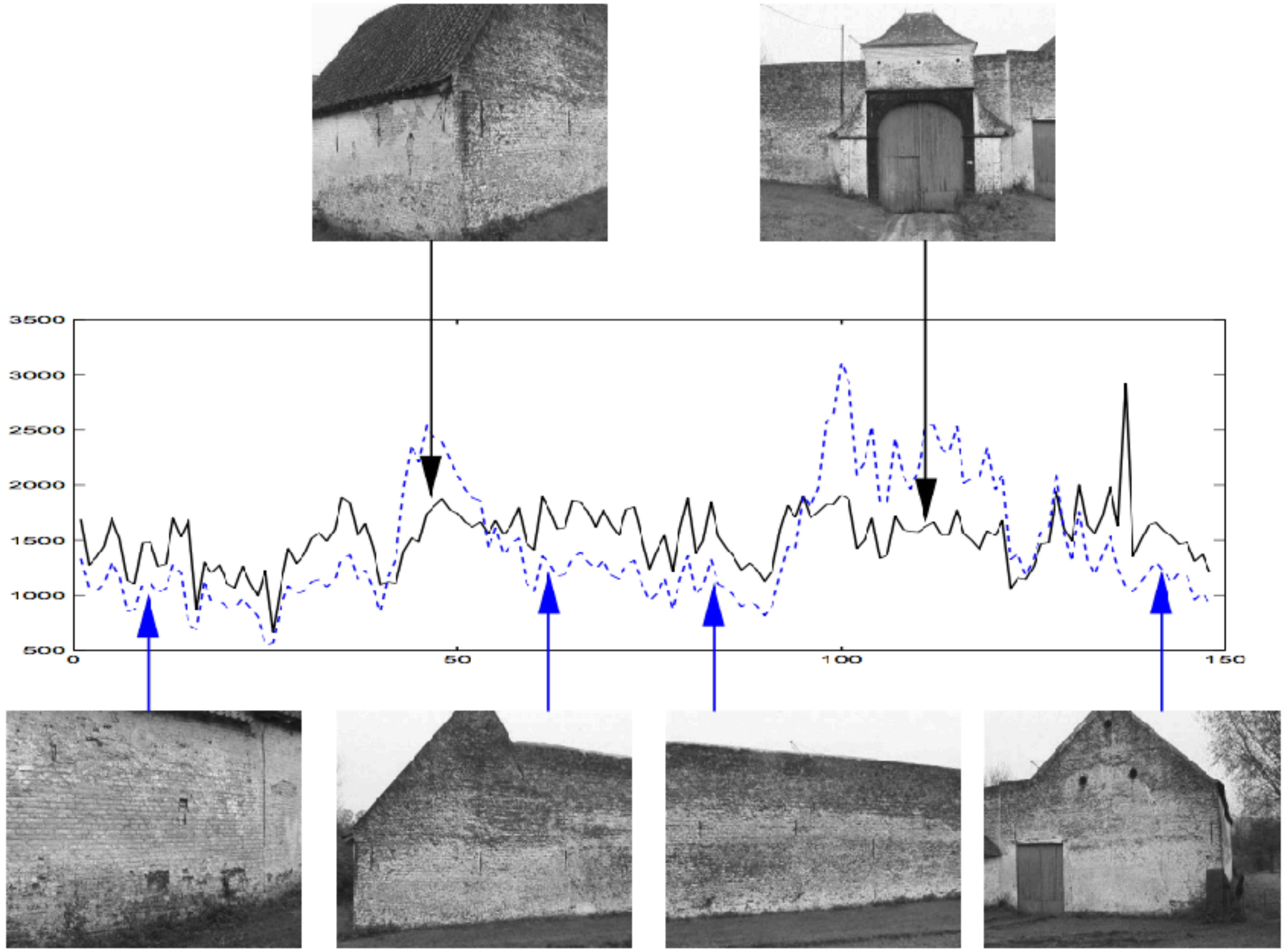


Figure 5.10: Some images of the *farmhouse* sequence together with GRIC(**PPP**) (solid/black line) and GRIC(**HH**) (dashed/blue line).

EXAMPLE

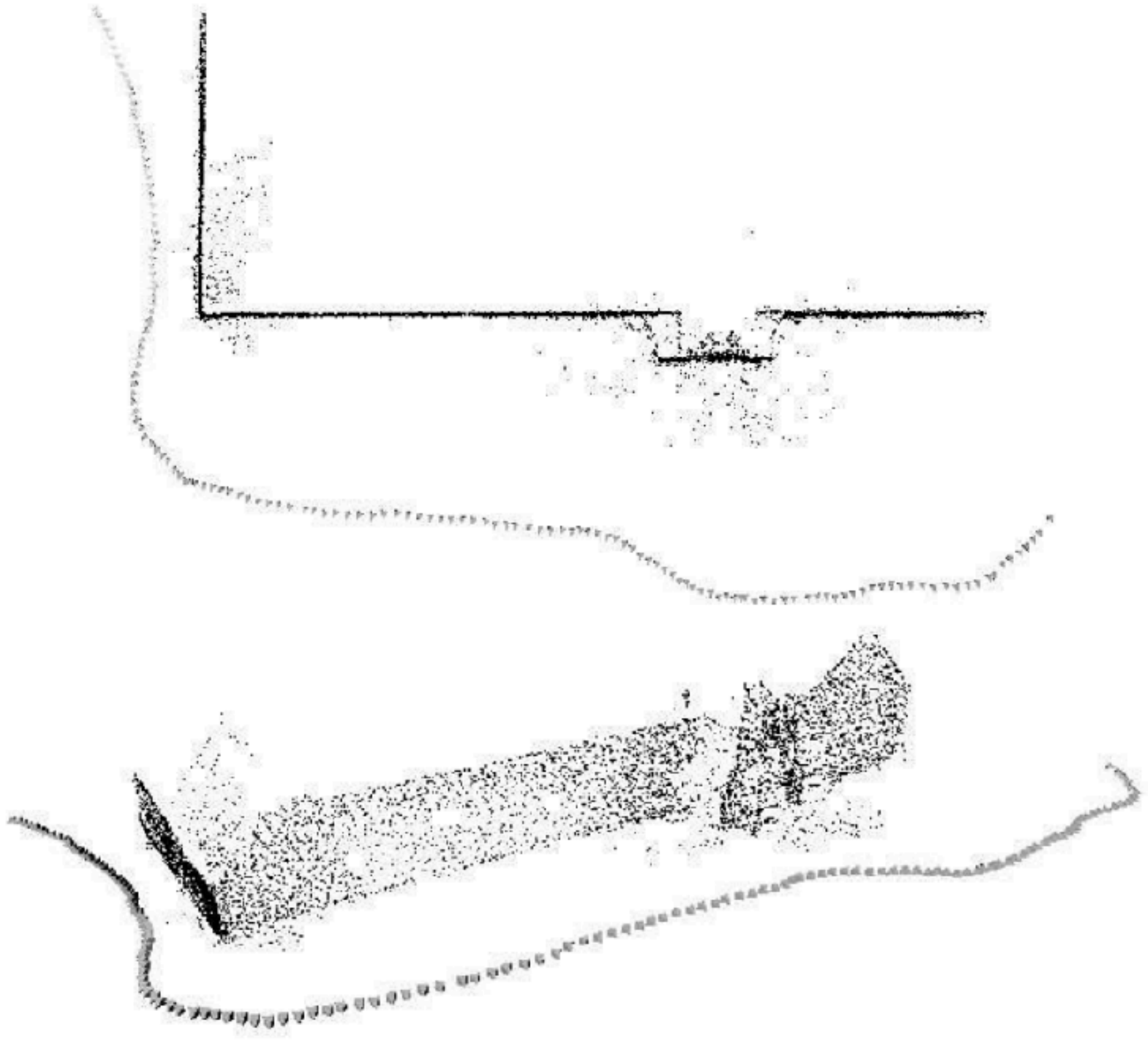


Figure 5.11: Combined structure and motion for the whole *farmhouse* sequence.

Figure 5.12: Textured 3D model of the *farmhouse*

DENSE MULTIVIEW

MULTI-VIEW STEREO

- small baseline stereo
  - we define viewpoints where the baseline is much smaller than the observed average scene depth. This configuration is usually valid for image sequences were the images are taken as a spatial sequence from many slightly varying view-points.
- wide baseline stereo
  - is used mostly with still image photographs of a scene where few images are taken from a very different viewpoint
  - quasi-dense
- multi-viewpoint linking
  - In addition it will produce denser depth maps than either of the other techniques, and allows additional features for depth and texture fusion

SMALL BASELINE STEREO

- + easy correspondence estimation, since the views are similar,
- + small regions of viewpoint related occlusions
- – small triangulation angle, hence large depth uncertainty.

WIDE BASELINE STEREO

- – hard correspondence estimation, since the views are not similar,
- – large regions of viewpoint related occlusions,
- + big triangulation angle, hence high depth accuracy

MULTI-VIEWPOINT LINKING STEREO

- + very dense depth maps for each viewpoint,
- + no viewpoint dependent occlusions,
- + highest depth resolution through viewpoint fusion,
- + texture enhancement (mean texture, highlight removal, super-resolution texture).

CORRESPONDENCE LINKING ALGORITHM

- It concatenates corresponding image points over multiple viewpoints by correspondence tracking over adjacent image pairs.
  - This of course implies that the individually measured pair matches are accurate.
  - To account for outliers in pair matches, some robust control strategies need to be employed to check the validity of the correspondence linking.

CORRESPONDENCE LINKING ALGORITHM

- Consider an image sequence taken from k = [1,N] viewpoints keeping the object in view.
- For any view point k let us consider the image triple [I_{k-1}, I_k , I_{k+1}]
- The image pairs (k-1, k) and (k , k+1) form two stereoscopic image pairs
- We have now defined 3 representations of image and camera matrices for each viewpoint:
  - the original image I_k and projection P_k matrix ,
  - their transformed versions I_k^{k-1}, P_k^{k-1} rectified towards view point k-1 with transformation R_k^{k-1} and
  - equivalent for K+1.
  - The Disparity map D(k,k-1) holds the downward correspondences from I_k^{k-1} to I_{k-1}^{k}

$$
\begin{aligned}
\text{Upwards linking:} \quad & \mathrm{m}_{k+1} = \left(\mathbf{R}_{k+1}^{k}\right)^{-1} D_{(k,k+1)}[\mathbf{R}_{k}^{k+1}\mathrm{m}_k] \\
\text{Downwards linking:} \quad & \mathrm{m}_{k-1} = \left(\mathbf{R}_{k-1}^{k}\right)^{-1} D_{(k,k-1)}[\mathbf{R}_{k}^{k-1}\mathrm{m}_k]
\end{aligned}
$$

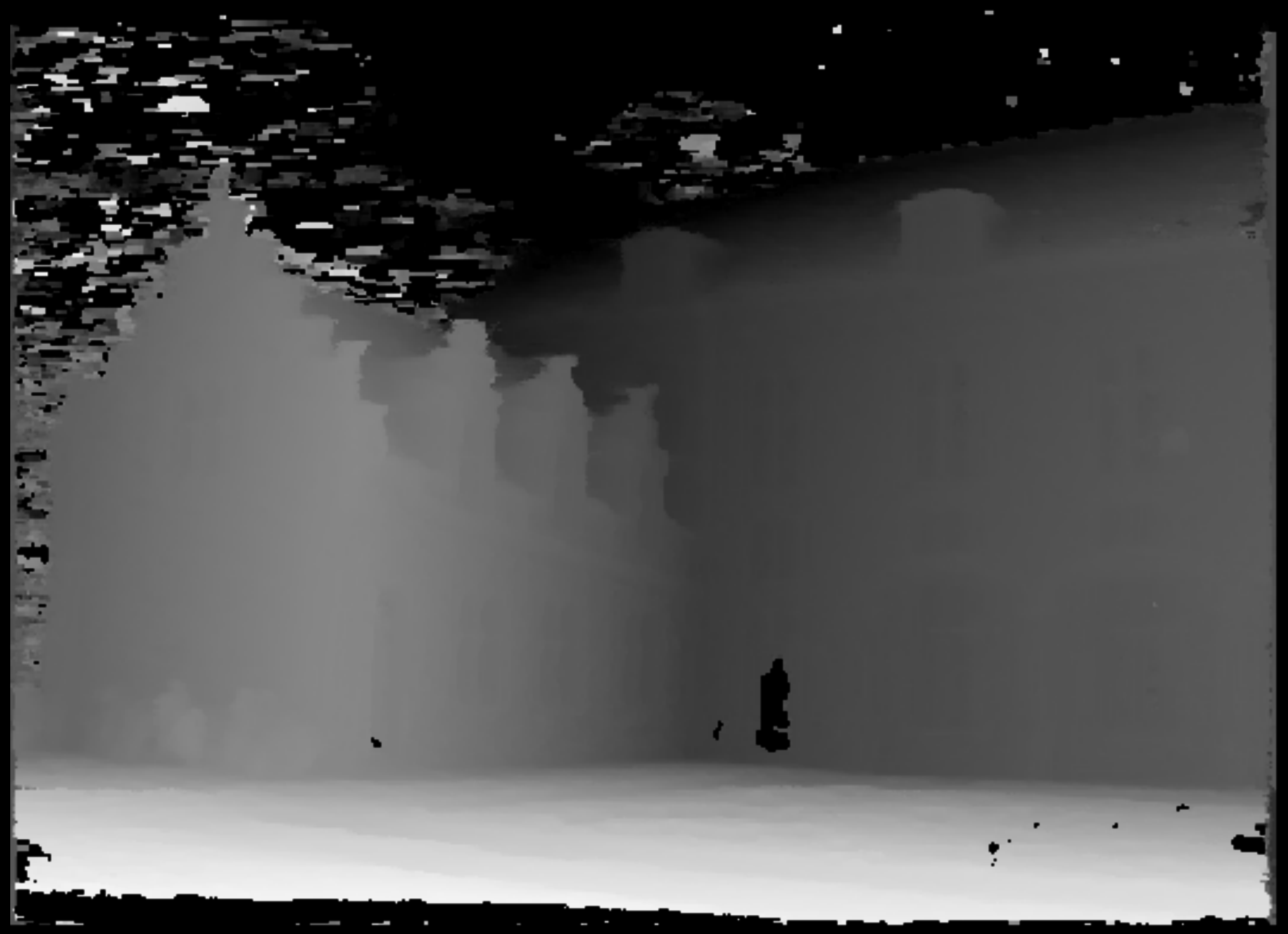
CORRESPONDENCE LINKING ALGORITHM

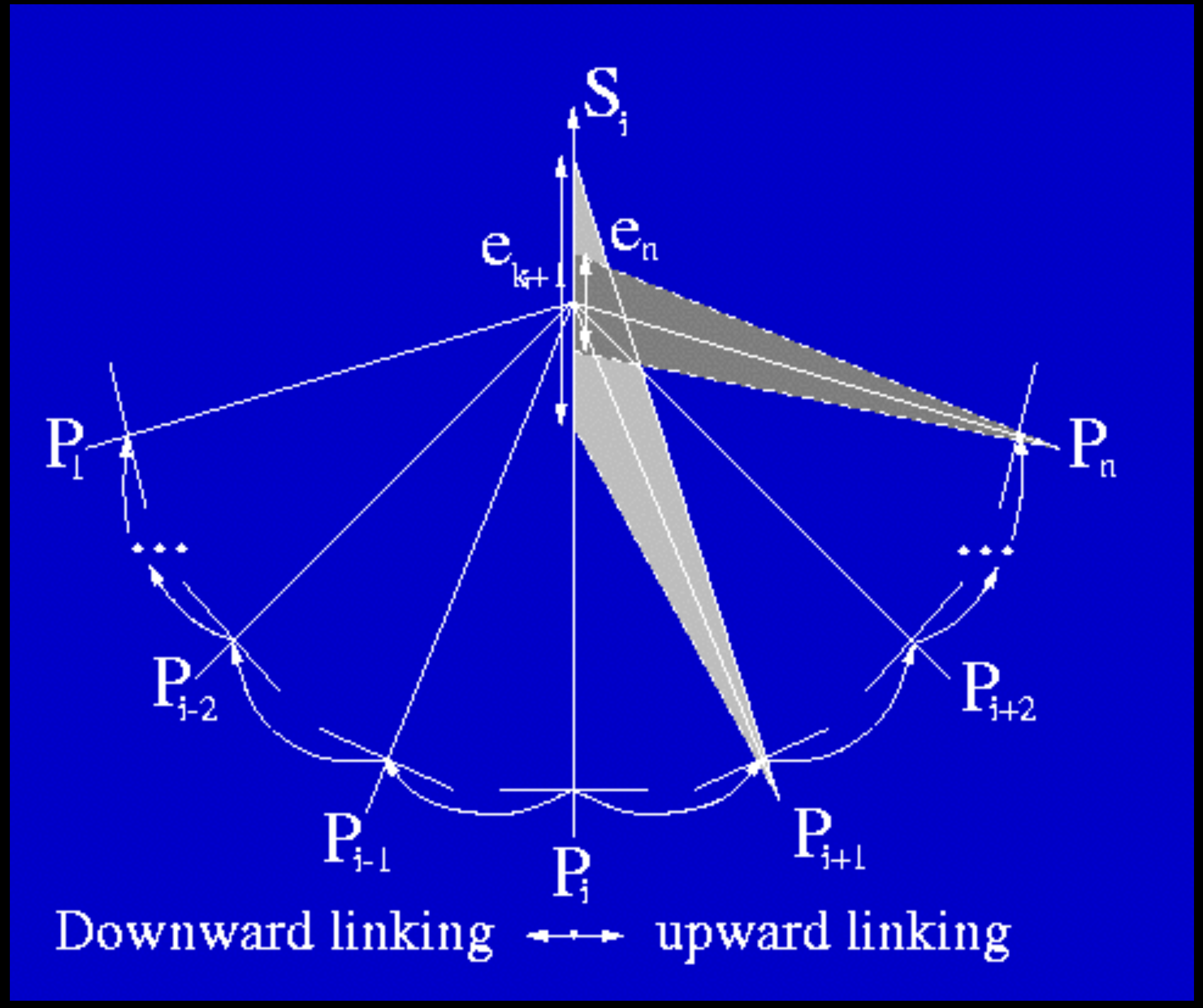
- This linking process is repeated along the image sequence to create a chain of correspondences upwards and downwards.
- Every correspondence link requires 2 mappings and 1 disparity lookup. Throughout the sequence of N images, 2 (N-1) disparity maps are computed. The multi-viewpoint linking is then performed efficiently via fast lookup functions on the pre-computed estimates.
- Due to the rectification mapping transformed image point will normally not fall on integer pixel coordinates in the rectified image. The lookup of an image disparity in the disparity map D will therefore require an interpolation function. Since disparity maps for piecewise continuous surfaces have a spatially low frequency content, a bilinear interpolation between pixels suffices

MULTI–VIEW DEPTH FUSION

(KOCH, POLLEFEYS AND VAN GOOL. ECCV'98)

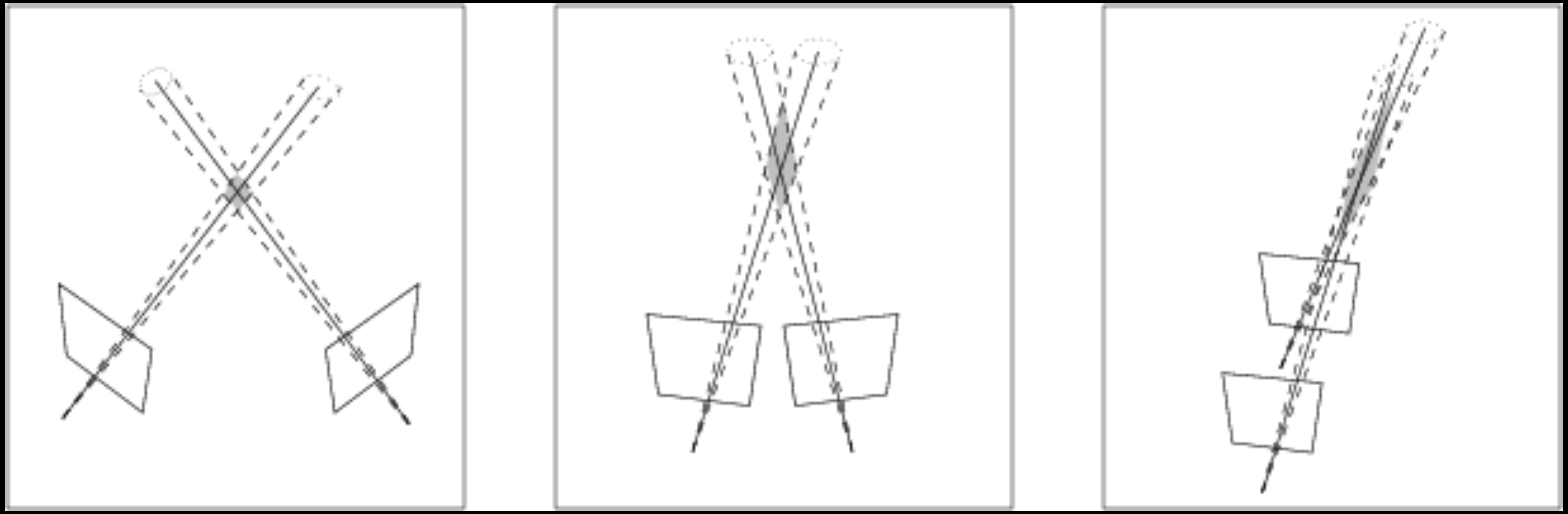
- Compute depth for every pixel of reference image
  - Triangulation
  - Use multiple views
  - Up- and down sequence
  - Use Kalman filter





ALLOWS TO COMPUTE ROBUST TEXTURE

RECONSTRUCTION UNCERTAINTY



CONSIDER ANGLE BETWEEN RAYS

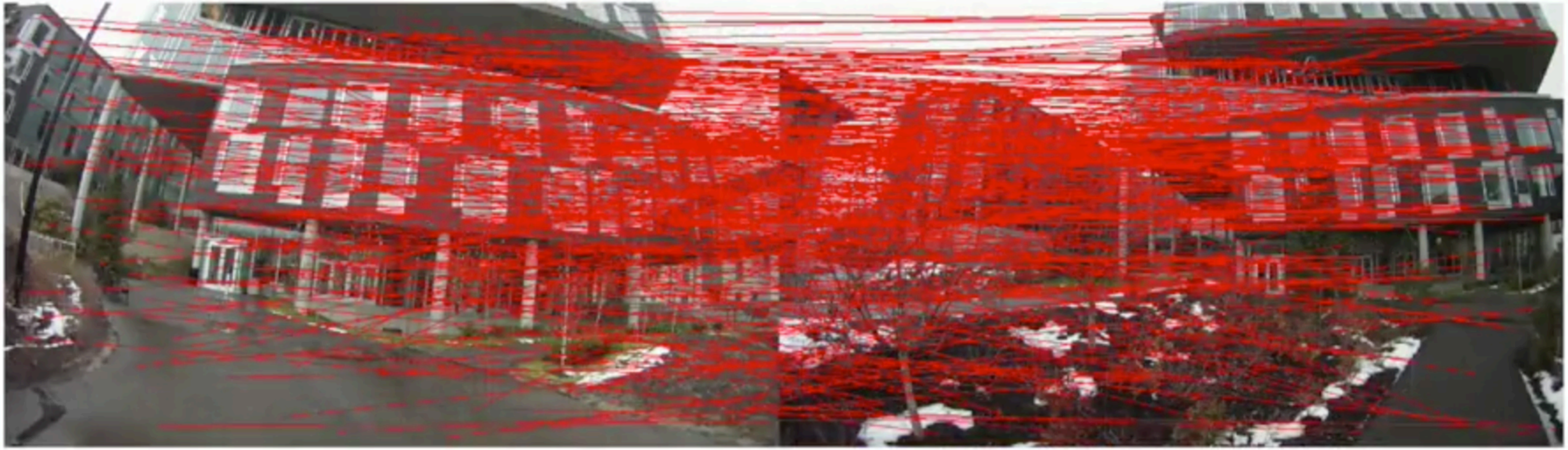
TEXT

TEXT

Initial Pair Images

TEXT

1. Pairwise Image Feature Matching



2. Outlier Rejection via RANSAC

$\mathbf{x}_1$ $\mathbf{x}_2$

$$\mathbf{x}_2^T F \mathbf{x}_1 = 0$$

Random sampling

Model building

Thresholding

Inlier counting

TEXT

2. Outlier Rejection via RANSAC

x_1 x_2

$$x_2^T F x_1 = 0$$

Random sampling

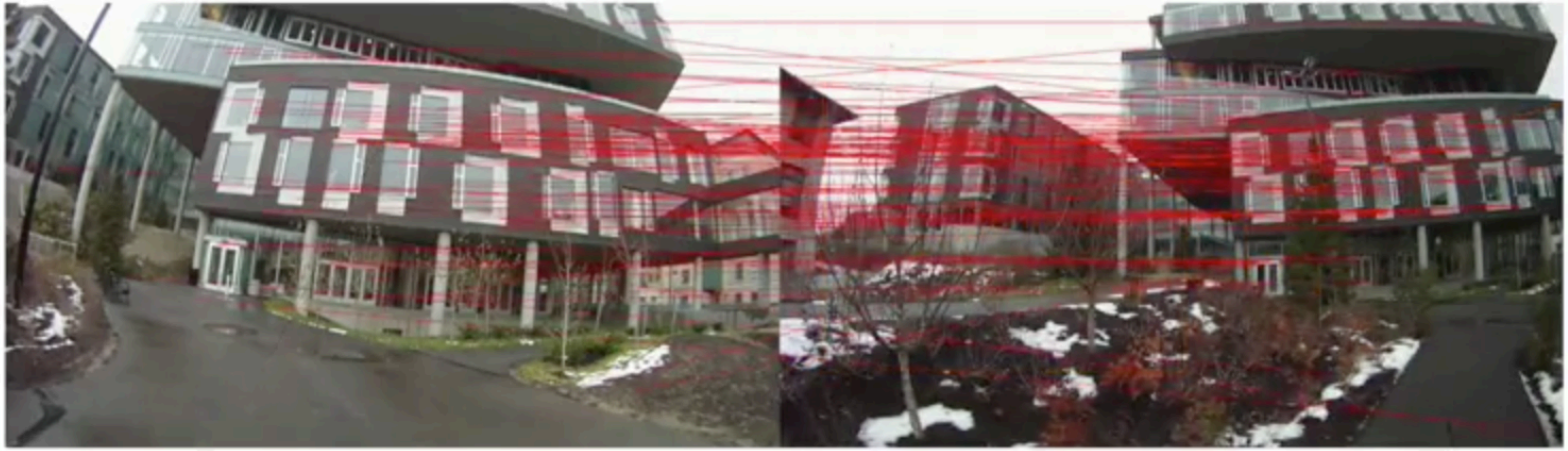
Model building

Thresholding

Inlier counting

TEXT

2. Outlier Rejection via RANSAC



$$x_2^T F x_1 = 0$$

\# of inliers: 253

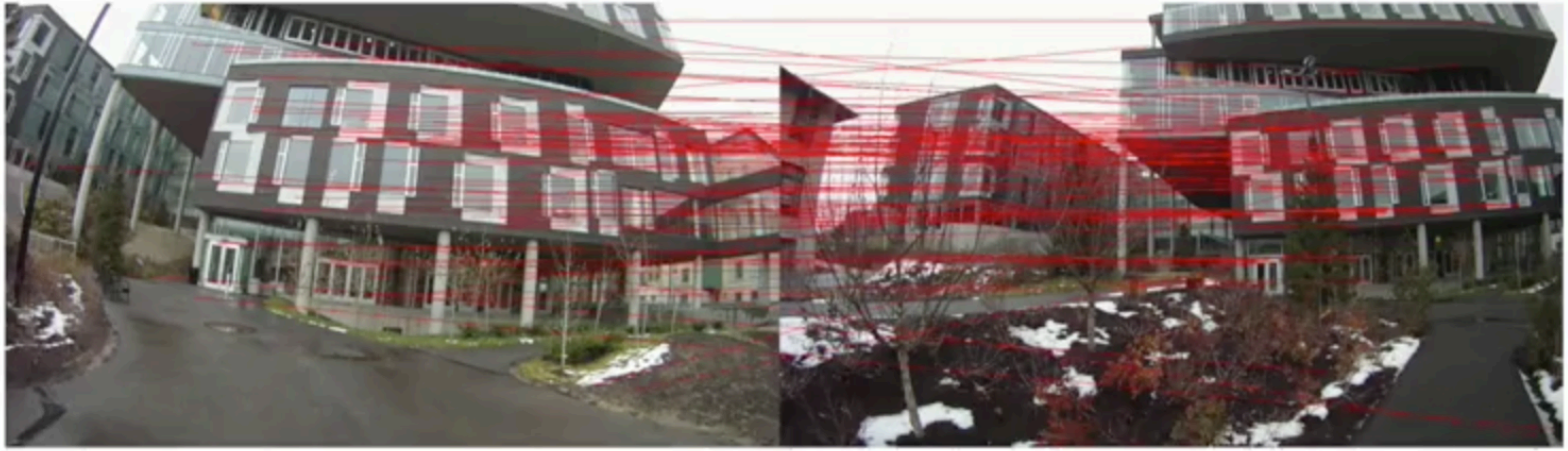
Random sampling

Model building

Thresholding

Inlier counting

4. Relative Transform from Essential Matrix

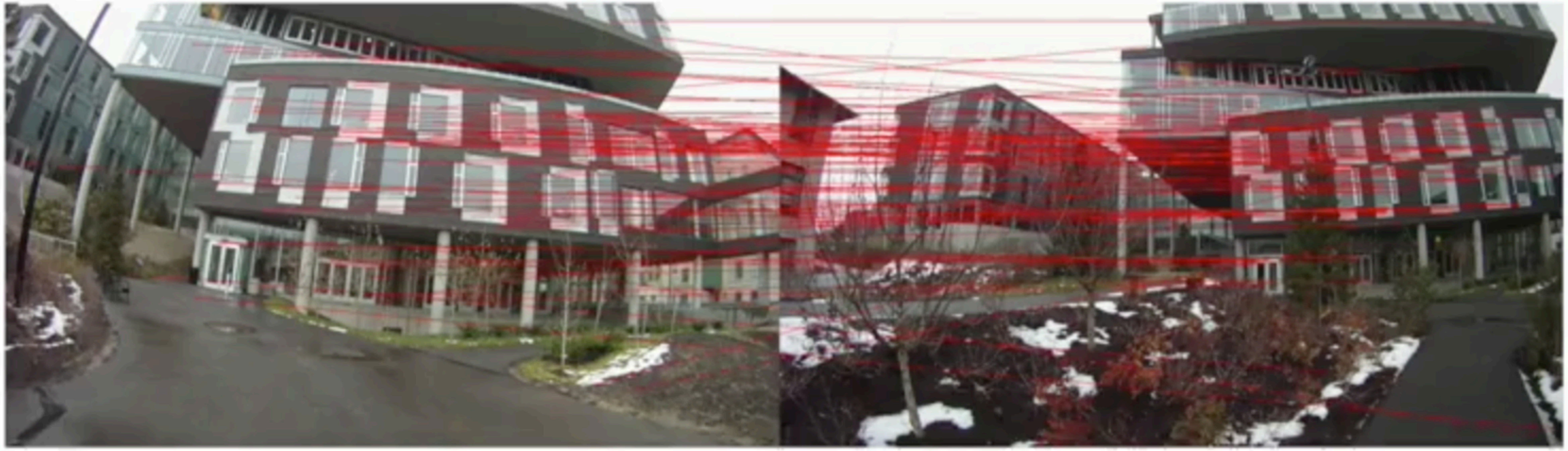


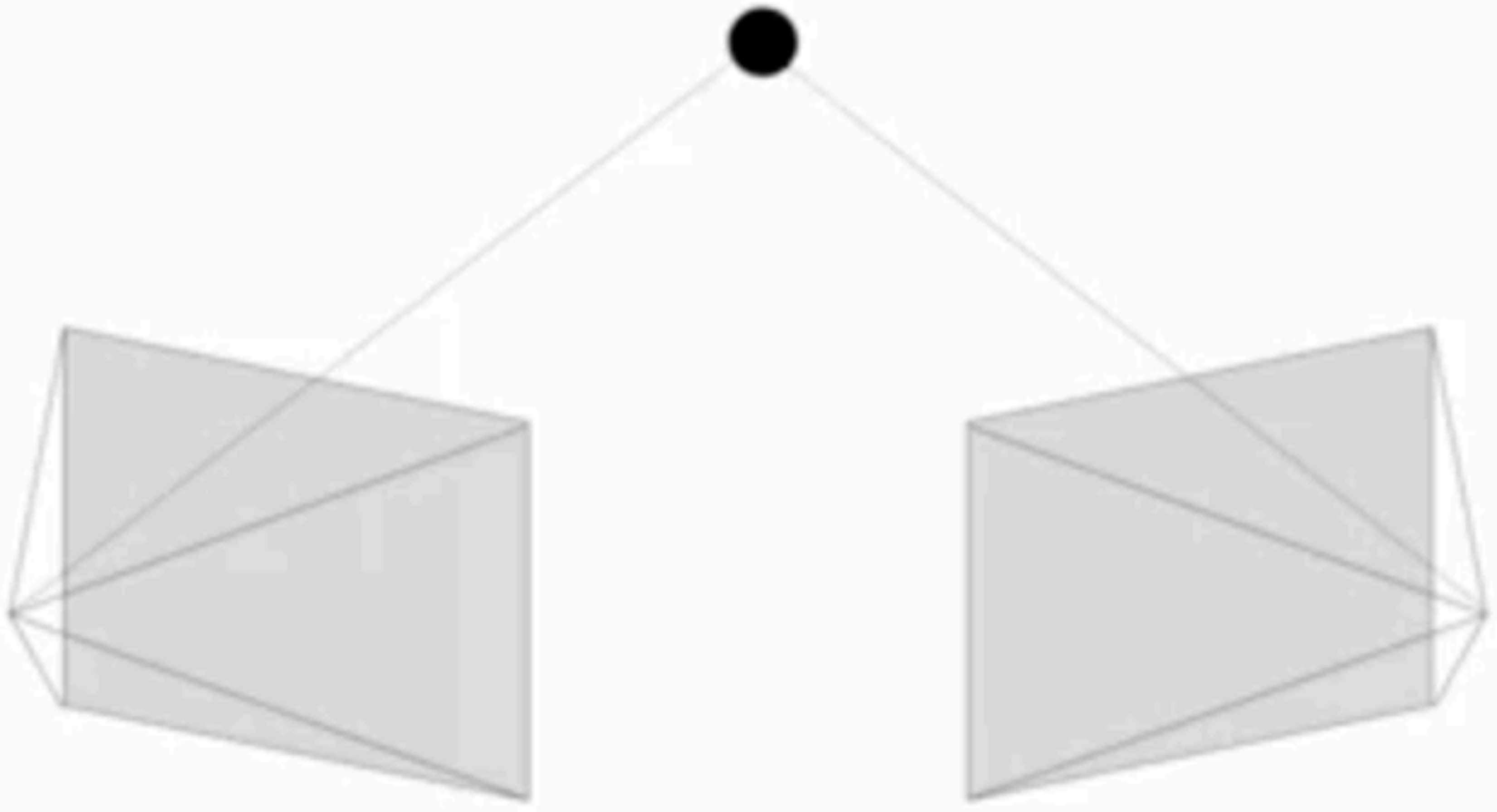
$$E = [t]_\times R$$

$$P_1 = [I_{3\times 3} \mid 0_3] \qquad P_2 = [R \mid t]$$

$$P_2 = [UYV^T \mid u_3]$$
$$[UY^TV^T \mid u_3]$$
$$[UYV^T \mid -u_3]$$
$$[UY^TV^T \mid -u_3]$$

5. Point Triangulation



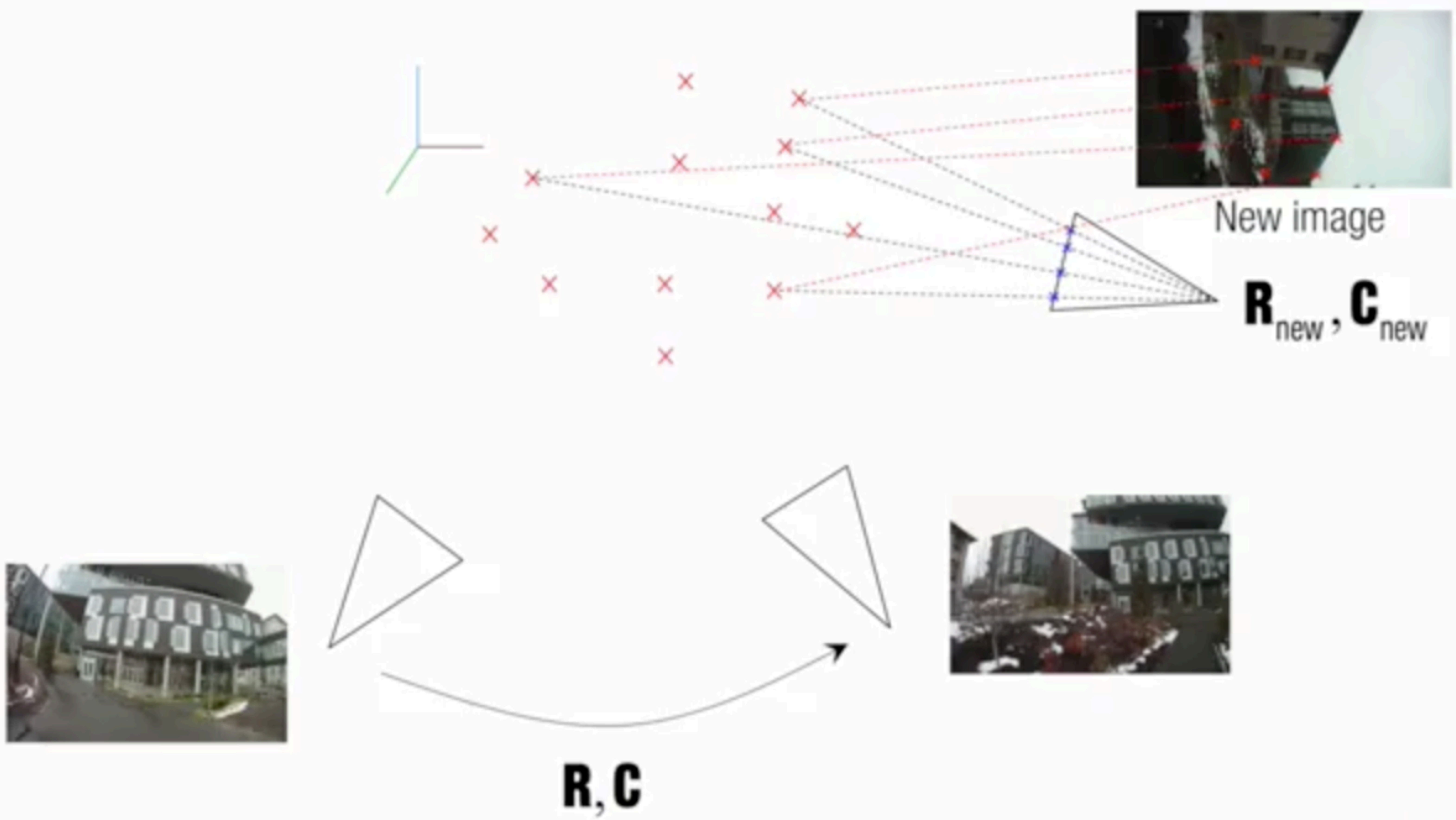


$$\begin{bmatrix} \begin{bmatrix} x_1 \\ 1 \end{bmatrix}_\times P_1 \\ \begin{bmatrix} x_2 \\ 1 \end{bmatrix}_\times P_2 \end{bmatrix} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

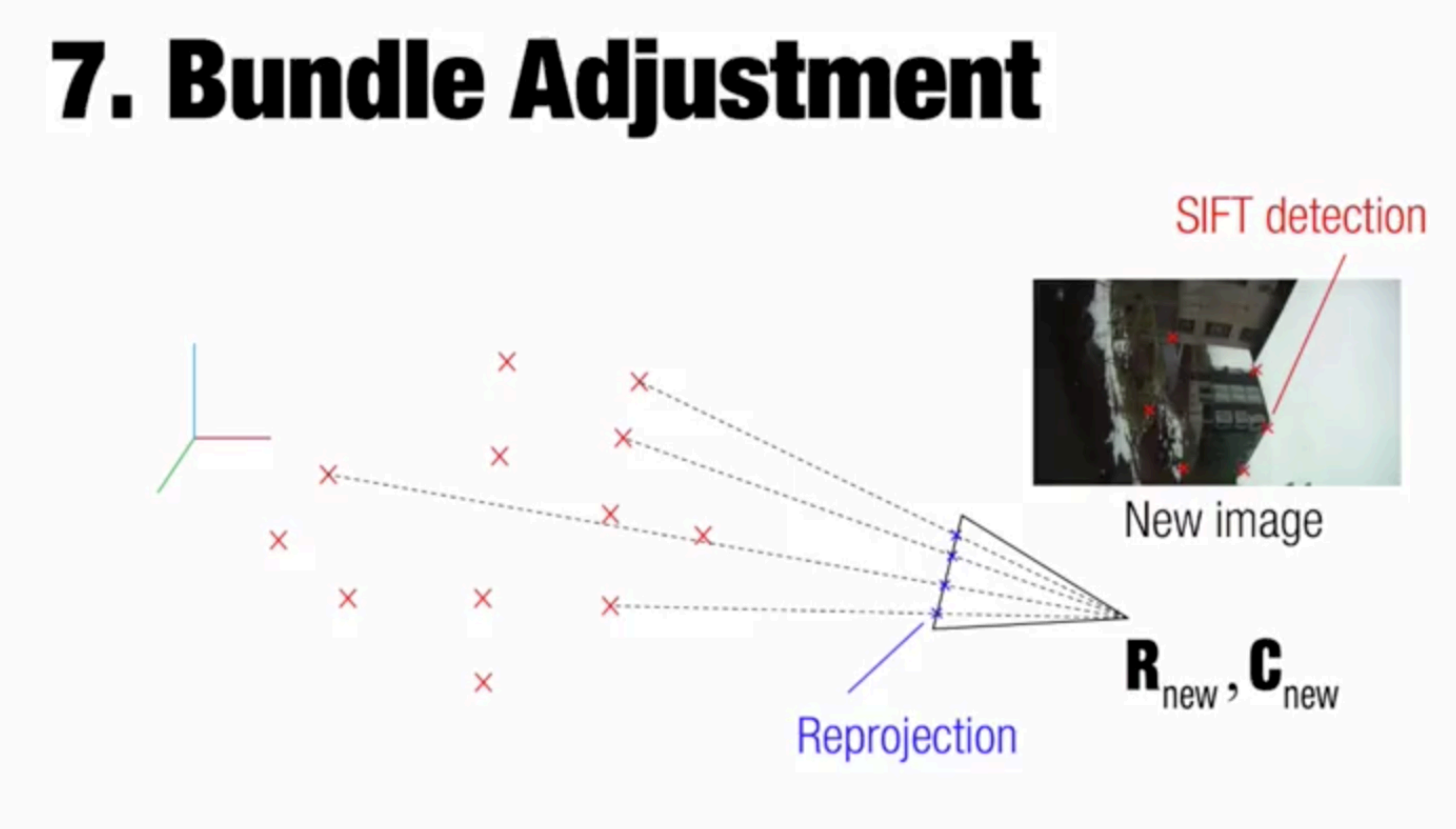
TEXT

6. New Camera Registration

Perspective-n-point



TEXT


7. Bundle Adjustment
SIFT detection
New image
R_{new}, C_{new}
Reprojection

- Some structures might need special treatment
- Some motions might need special auto calibration

SOURCES

- https://www.coursera.org/learn/robotics-perception/lecture/oDj0o/bundle-adjustment-i
- http://www.cs.unc.edu/~marc/tutorial/