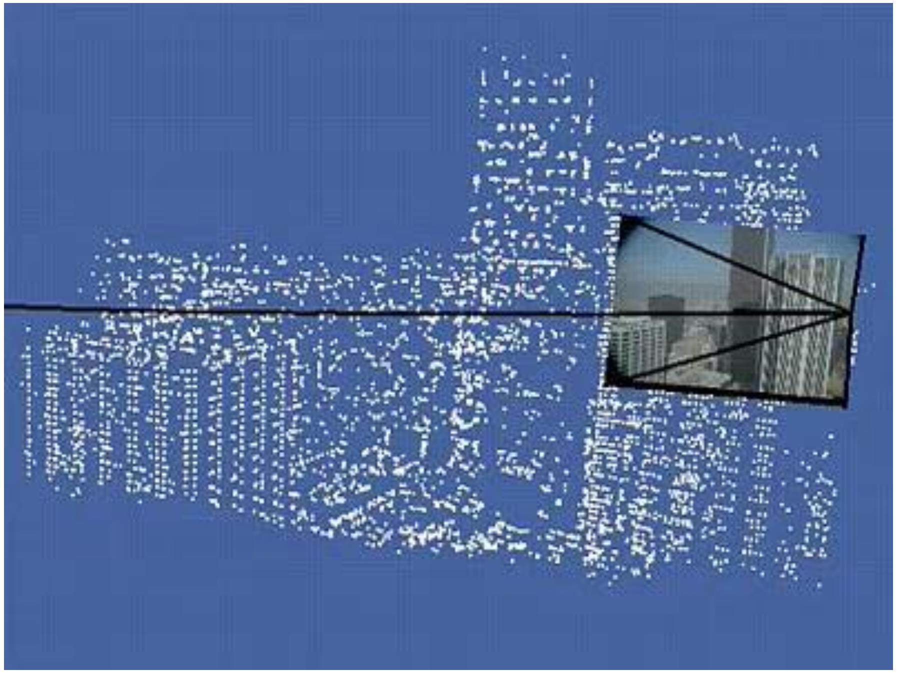
Lecture 7
Multi-view geometry

- The SFM problem
- Affine SFM
- Perspective SFM
- Self-calibration

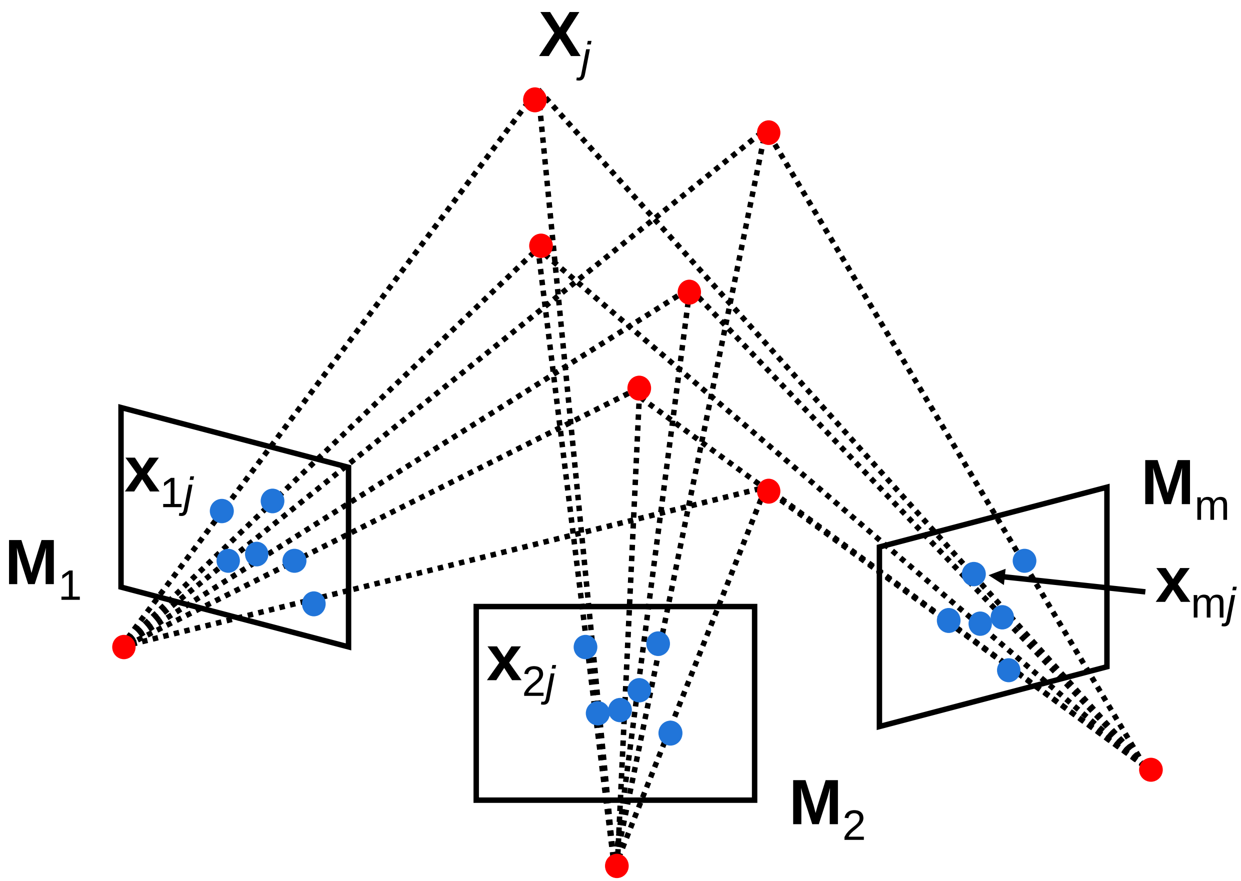
Reading:
[HZ] Chapters: 10 “3D reconstruction of cameras and structure”,
18 “N-view computational methods”,
19 “Auto-calibration”
[FP] Chapter 13: “projective structure from motion”13
[Szelisky] Chapter 7 “Structure from motion”

Structure from motion problem



Courtesy of Oxford **Visual Geometry Group**

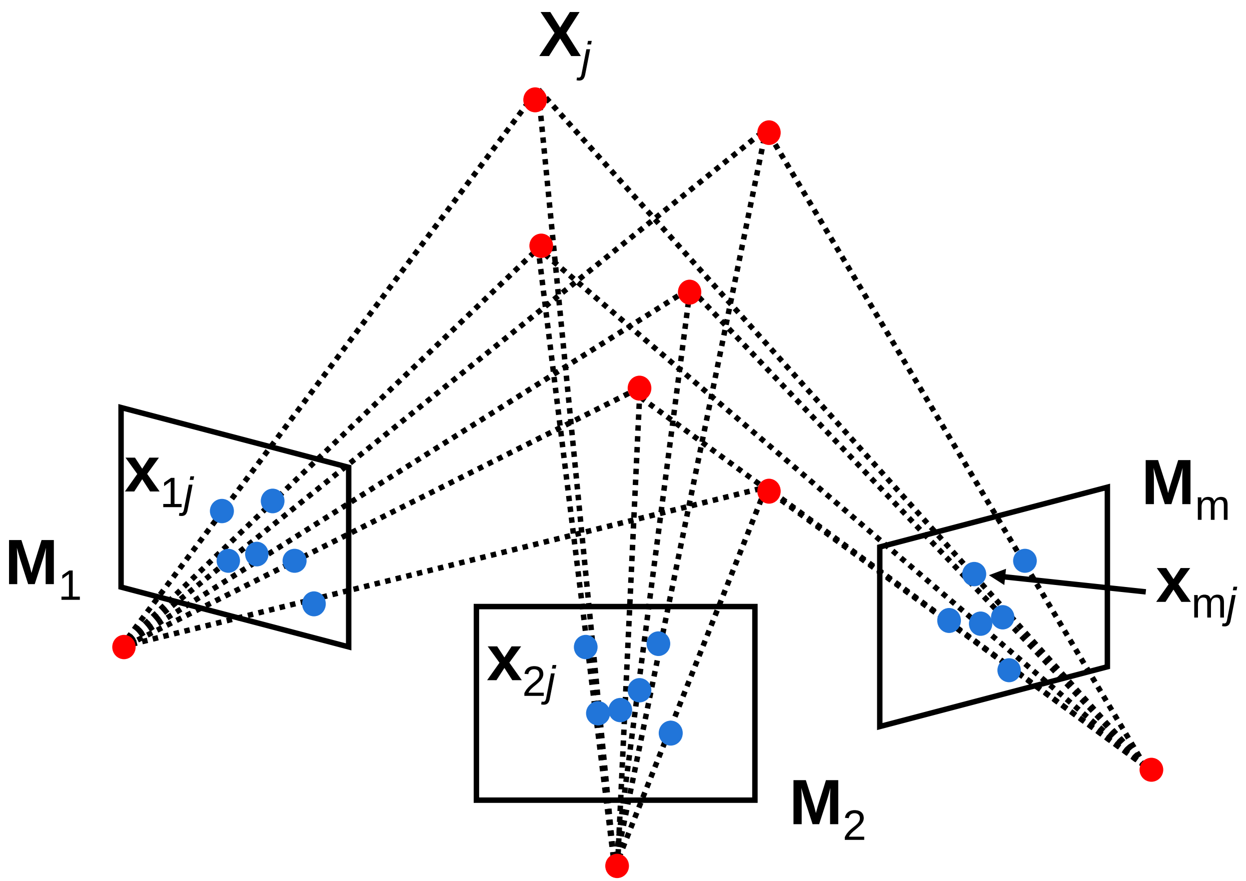
Structure from motion problem



Given *m* images of *n* fixed 3D points

- $\mathbf{x}_{ij} = \mathbf{M}_i \mathbf{X}_j , \quad i = 1, \ldots , m, \quad j = 1, \ldots , n$

Structure from motion problem



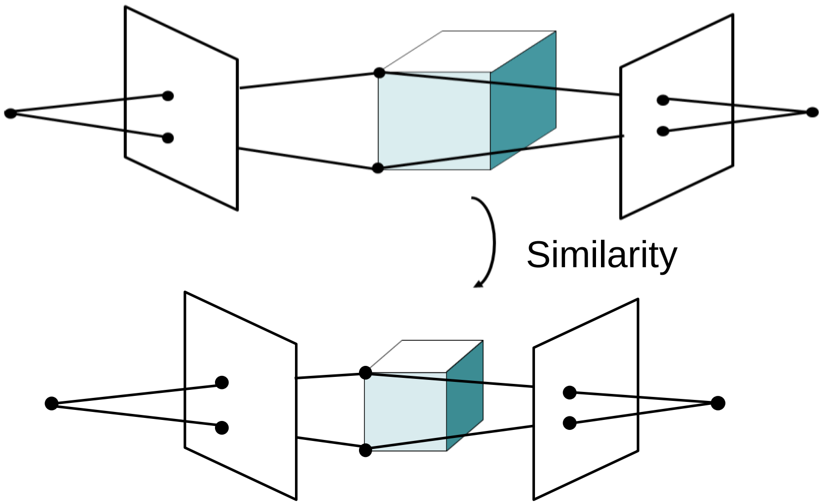
From the m×n correspondences $\mathbf{x}_{ij}$, estimate:

- m projection matrices $\mathbf{M}_i$ — motion
- n 3D points $\mathbf{X}_j$ — structure

Similarity Ambiguity

- The scene is determined by the images only up a similarity transformation (rotation, translation and scaling)

- This is called **metric reconstruction**

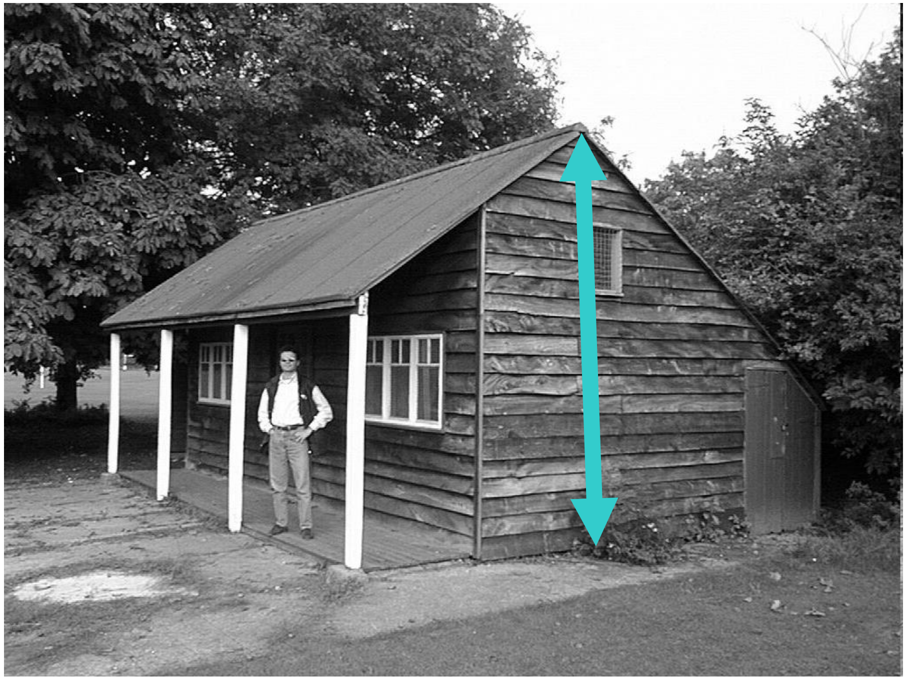


- The ambiguity exists even for (intrinsically) calibrated cameras
- For calibrated cameras, the similarity ambiguity is the only ambiguity

[Longuet-Higgins '81]

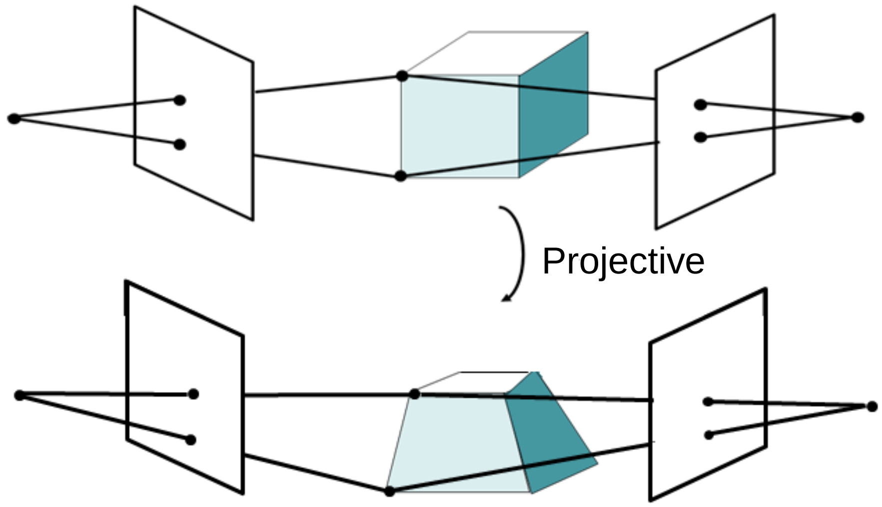
Similarity Ambiguity

- It is impossible based on the images alone to estimate the absolute scale of the scene (i.e. house height)



http://www.robots.ox.ac.uk/~vgg/projects/SingleView/models/hut/hutme.wrl

Structure from Motion Ambiguities



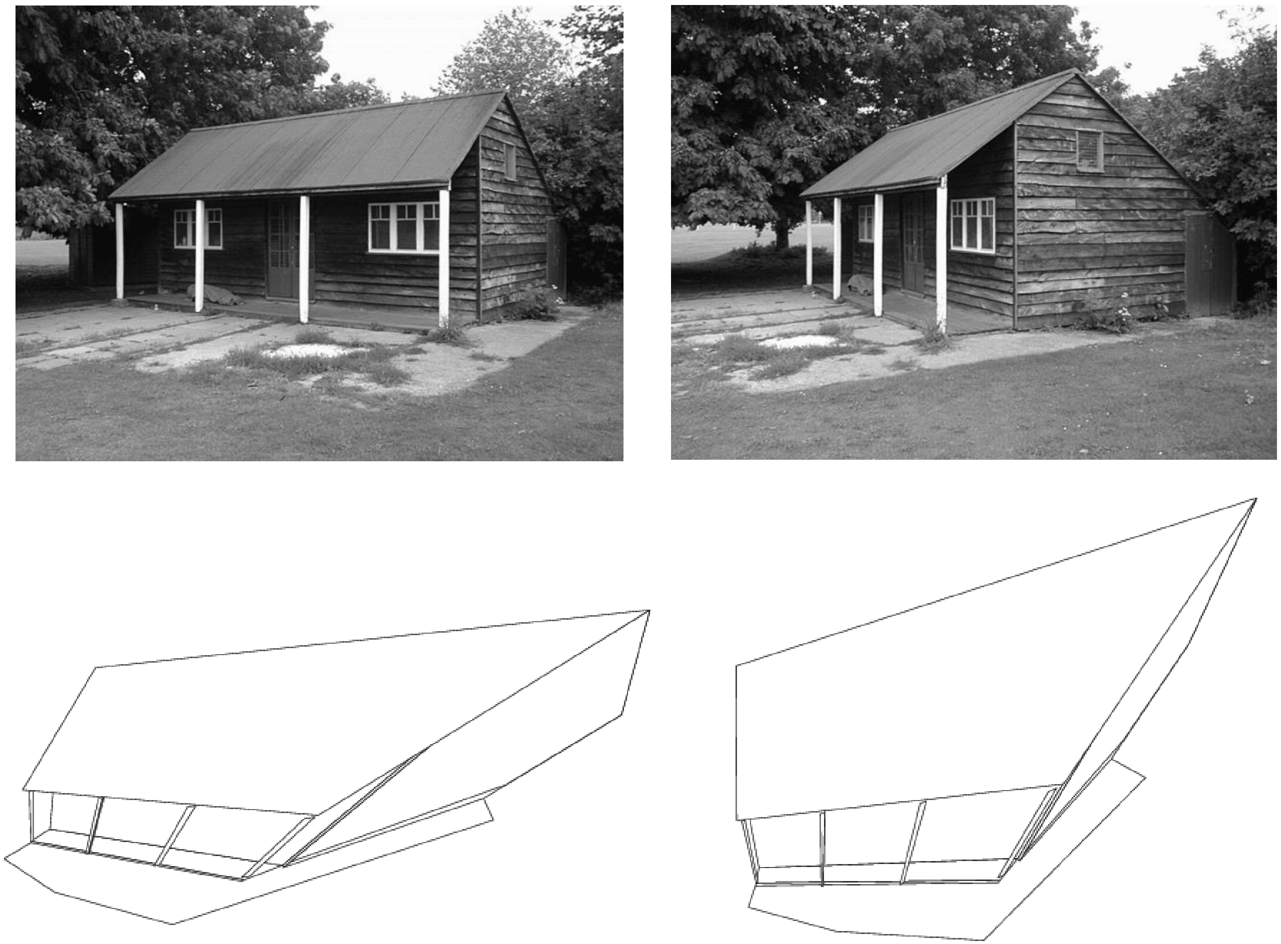
- In the general case (nothing is known) the ambiguity is expressed by an arbitrary affine or projective transformation

$$x_j = M_i X_j \qquad M_i = K_i [R_i \quad T_i]$$

$$X_j \rightarrow H X_j \qquad M_i \rightarrow M_j H^{-1}$$

$$x_j = M_i X_j = (M_i H^{-1})(H X_j)$$

Projective Ambiguity

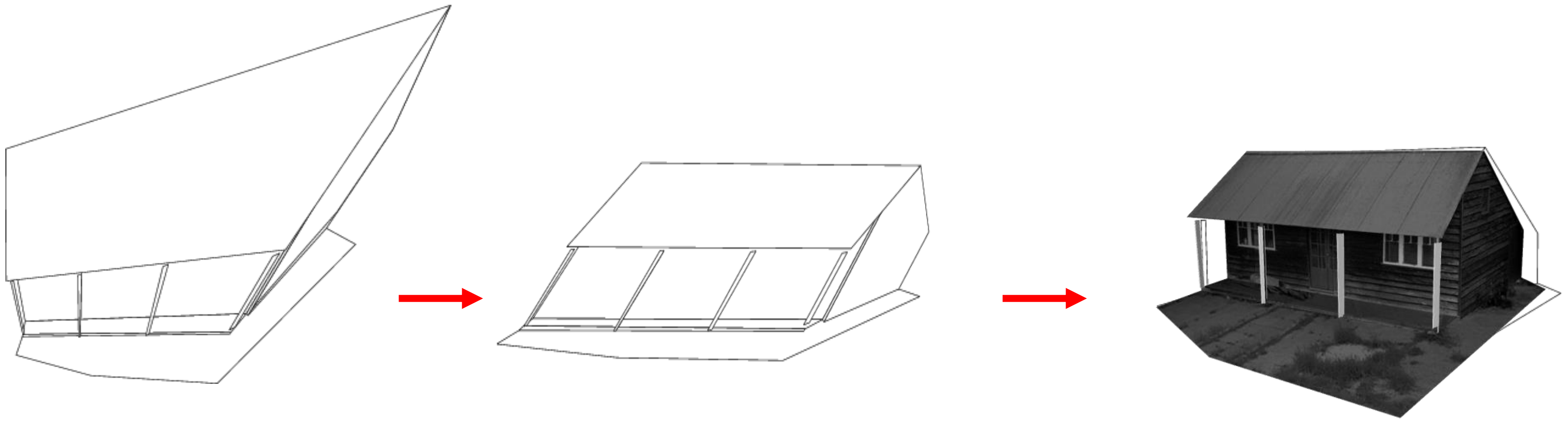


R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, 2nd edition, 2003

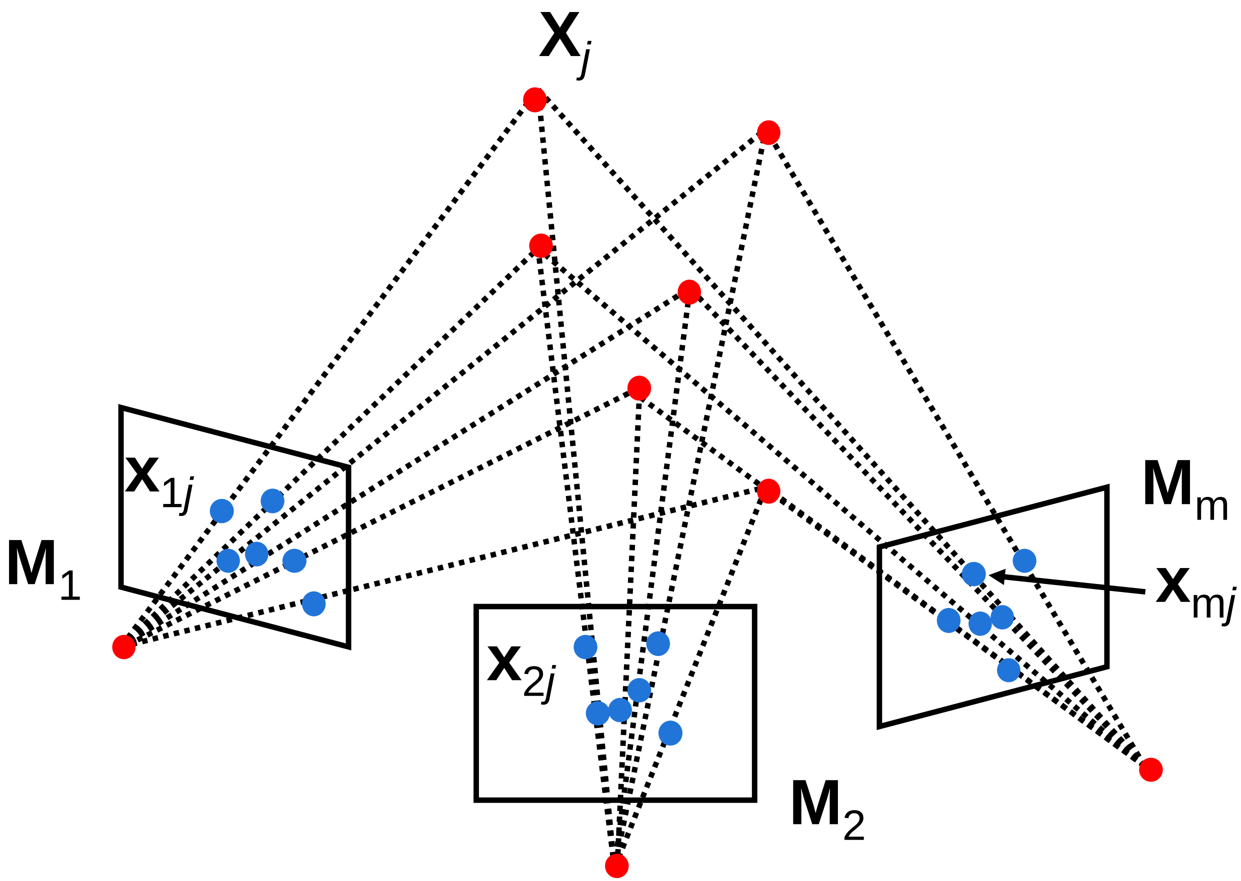
Metric reconstruction (upgrade)

- The problem of recovering the metric reconstruction from the perspective one is called self-calibration

- Stratified reconstruction:
  - from perspective to affine
  - from affine to metric



Structure from motion problem



From the mxn correspondences $\mathbf{x}_{ij}$, estimate:

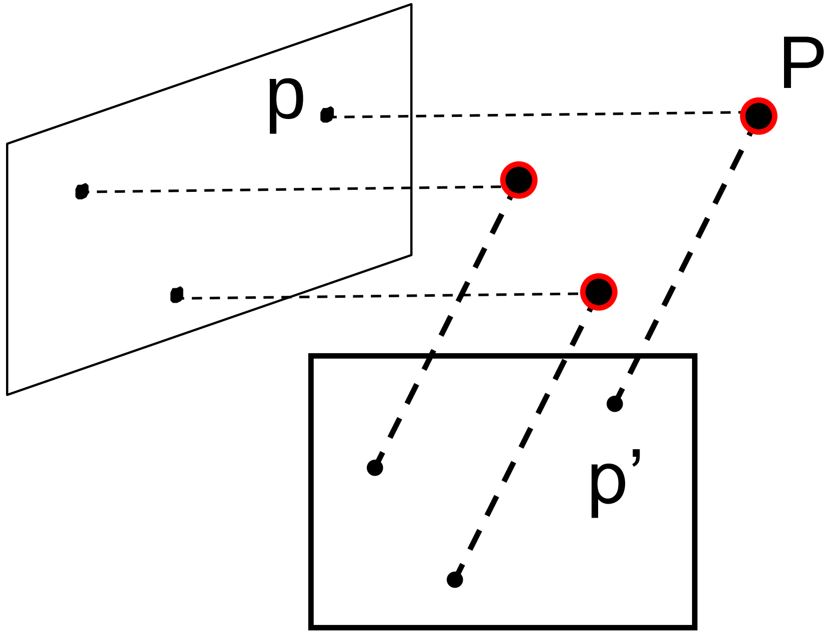
- m projection matrices $\mathbf{M}_i$
- n 3D points $\mathbf{X}_j$
- Upgrade to metric reconstruction (self-calibration)

Lecture 6

Multi-view geometry

- The SFM problem
- Affine SFM
- Perspective SFM
- Self-calibration

Affine cameras

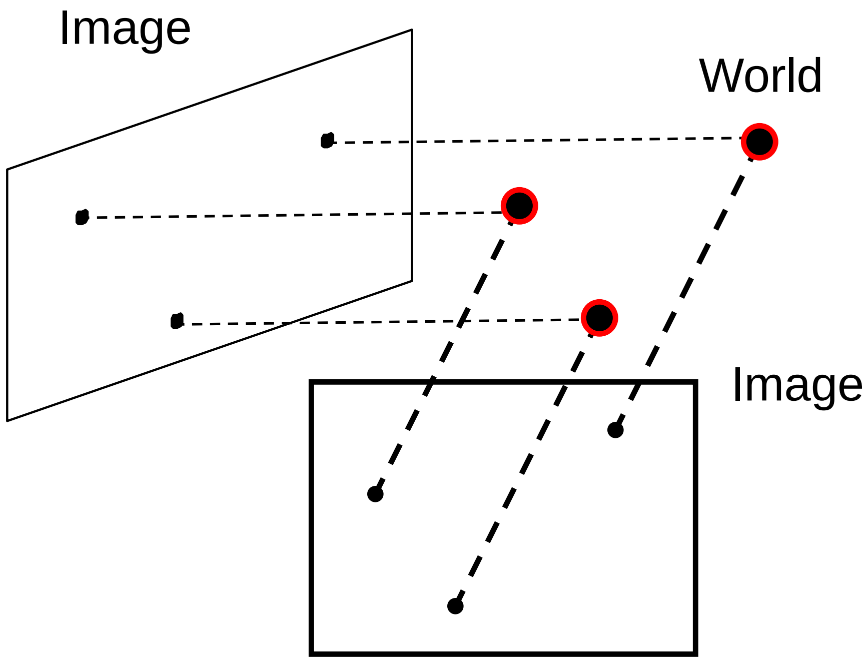


Camera matrix M for the affine case

$$\mathbf{p} = \begin{pmatrix} u \\ v \end{pmatrix} = \mathbf{AP} + \mathbf{b} = M \begin{bmatrix} \mathbf{P} \\ 1 \end{bmatrix}; \qquad \mathrm{M} = \begin{bmatrix} \mathrm{A} & \mathrm{b} \end{bmatrix}$$

Affine structure from motion

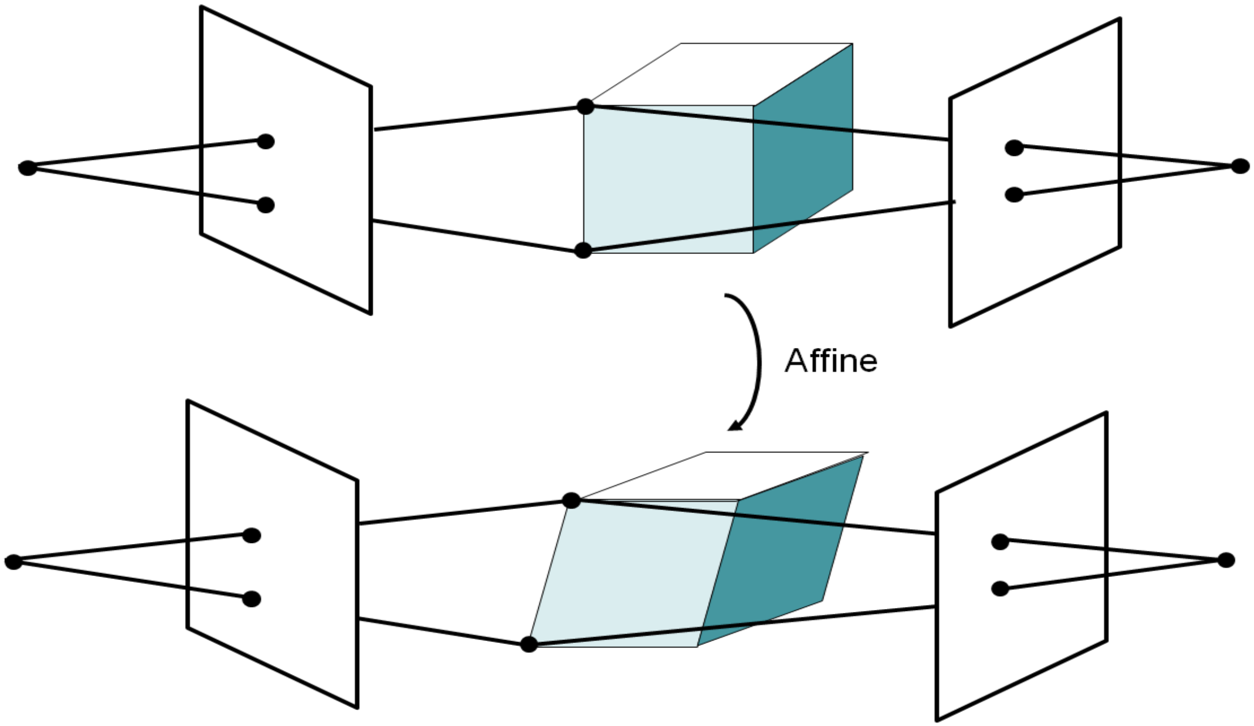
(simpler problem)



From the m x n correspondences $\mathbf{x}_{ij}$, estimate:

- *m* projection matrices $\mathbf{M}_i$ (affine cameras)
- *n* 3D points $\mathbf{X}_j$
- Upgrade to metric reconstruction

Affine Ambiguity



$$\mathbf{p} = \mathbf{M}\,\mathbf{P} = \left(\mathbf{M}\,\mathbf{Q}_A^{-1}\right)\left(\mathbf{Q}_A\,\mathbf{P}\right)$$

The Affine Structure-from-Motion Problem

Given m images of n fixed points P_j (=X_j) we can write

$$\mathbf{p}_{ij} = \mathrm{M}_i \begin{bmatrix} \mathbf{P}_j \\ 1 \end{bmatrix} = \mathbf{A}_i \mathbf{P}_j + \mathbf{b}_i \quad \text{for } i = 1, \ldots, m \text{ and } j = 1, \ldots, n.$$

M of cameras   N of points

Problem: estimate the m 2×4 matrices M_i and the n positions P_j from the m×n correspondences $\mathbf{p}_{ij}$.

How many equations and how many unknown?

2m × n equations in (8m+3n- 8) unknowns

Two approaches:

- Algebraic approach (affine epipolar geometry; estimate F; cameras; points)
- Factorization method

A factorization method – Tomasi & Kanade algorithm

C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *IJCV*, 9(2):137-154, November 1992.

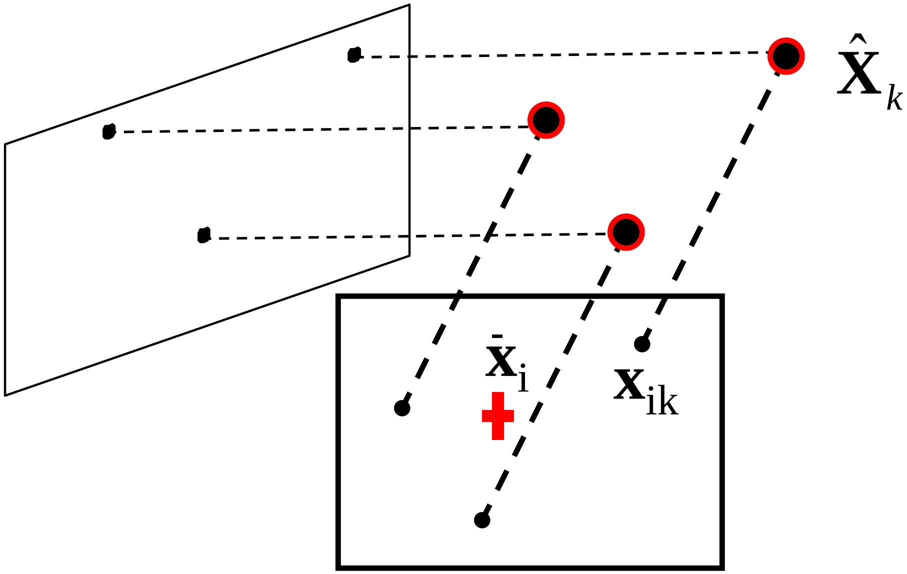
- Centering the data
- Factorization

A factorization method - Centering the data

Centering: subtract the centroid of the image points

$$\hat{\mathbf{x}}_{ij} = \mathbf{x}_{ij} - \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_{ik}$$

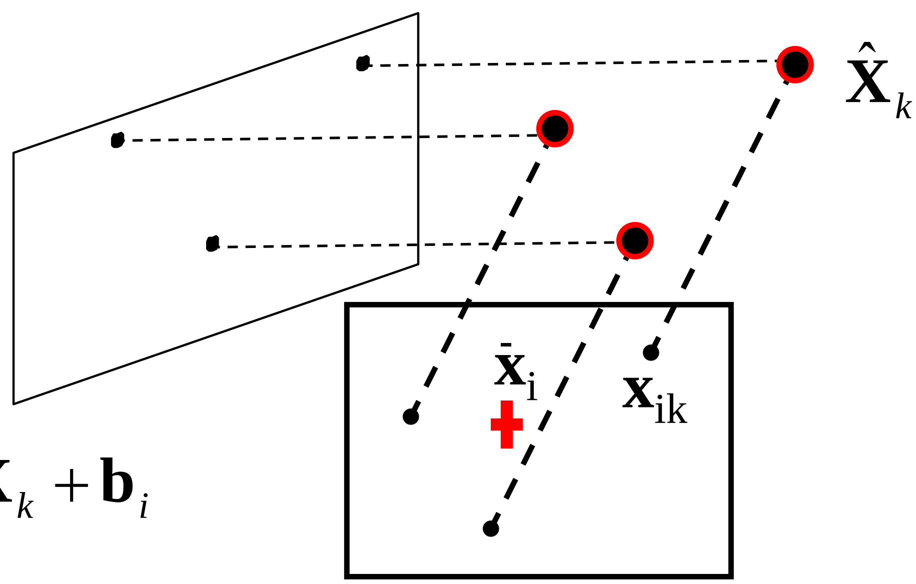
$\bar{\mathbf{x}}_i$



A factorization method - Centering the data

Centering: subtract the centroid of the image points

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



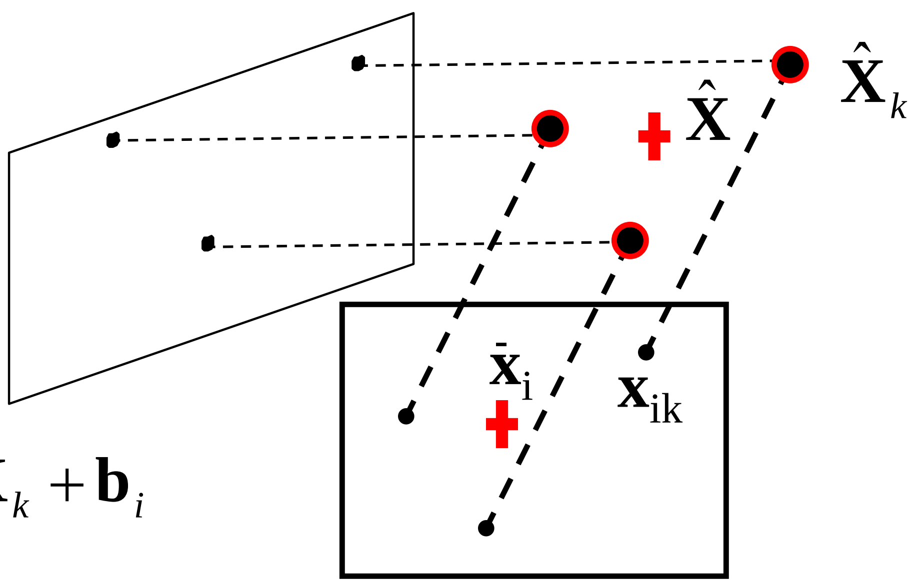
$$\mathbf{x}_{ik} = \mathbf{A}_i\mathbf{X}_k + \mathbf{b}_i$$

A factorization method - Centering the data

Centering: subtract the centroid of the image points

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

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



$$\mathbf{x}_{ik} = \mathbf{A}_i\mathbf{X}_k + \mathbf{b}_i$$

A factorization method - Centering the data

Centering: subtract the centroid of the image points

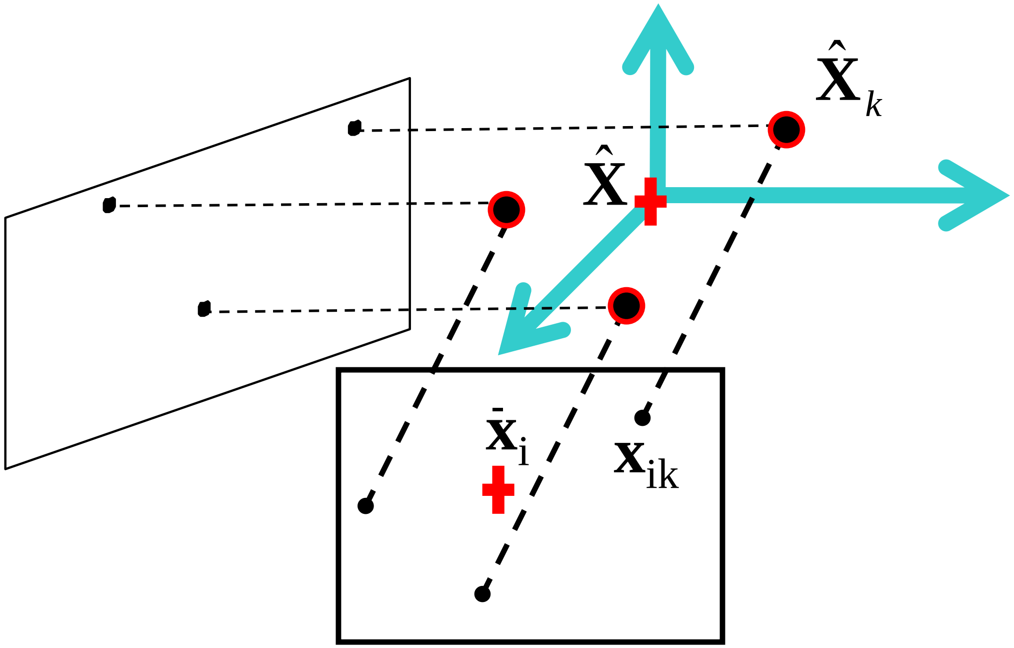
$$\hat{\mathbf{x}}_{ij} = \mathbf{x}_{ij} - \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_{ik} \quad = \mathbf{A}_i\mathbf{X}_j + \mathbf{b}_i - \frac{1}{n}\sum_{k=1}^{n}\left(\mathbf{A}_i\mathbf{X}_k + \mathbf{b}_i\right)$$

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

After centering, each normalized point $\mathbf{x}_{ij}$ is related to the 3D point $\mathbf{X}_i$ by

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

A factorization method - Centering the data



Centroid of points in 3D = center of the world reference system

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

A factorization method - factorization

Let's create a 2m × n data (measurement) matrix:

$$\mathbf{D} = \begin{bmatrix} \hat{\mathbf{x}}_{11} & \hat{\mathbf{x}}_{12} & \cdots & \hat{\mathbf{x}}_{1n} \\ \hat{\mathbf{x}}_{21} & \hat{\mathbf{x}}_{22} & \cdots & \hat{\mathbf{x}}_{2n} \\ & & \ddots & \\ \hat{\mathbf{x}}_{m1} & \hat{\mathbf{x}}_{m2} & \cdots & \hat{\mathbf{x}}_{mn} \end{bmatrix}$$

cameras ($2m$)

points (n)

A factorization method - factorization

Let's create a 2m × n data (measurement) matrix:

$$\mathbf{D} = \begin{bmatrix} \hat{\mathbf{x}}_{11} & \hat{\mathbf{x}}_{12} & \cdots & \hat{\mathbf{x}}_{1n} \\ \hat{\mathbf{x}}_{21} & \hat{\mathbf{x}}_{22} & \cdots & \hat{\mathbf{x}}_{2n} \\ & & \ddots & \\ \hat{\mathbf{x}}_{m1} & \hat{\mathbf{x}}_{m2} & \cdots & \hat{\mathbf{x}}_{mn} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_m \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \cdots & \mathbf{X}_n \end{bmatrix}$$

(2*m* × n) M S

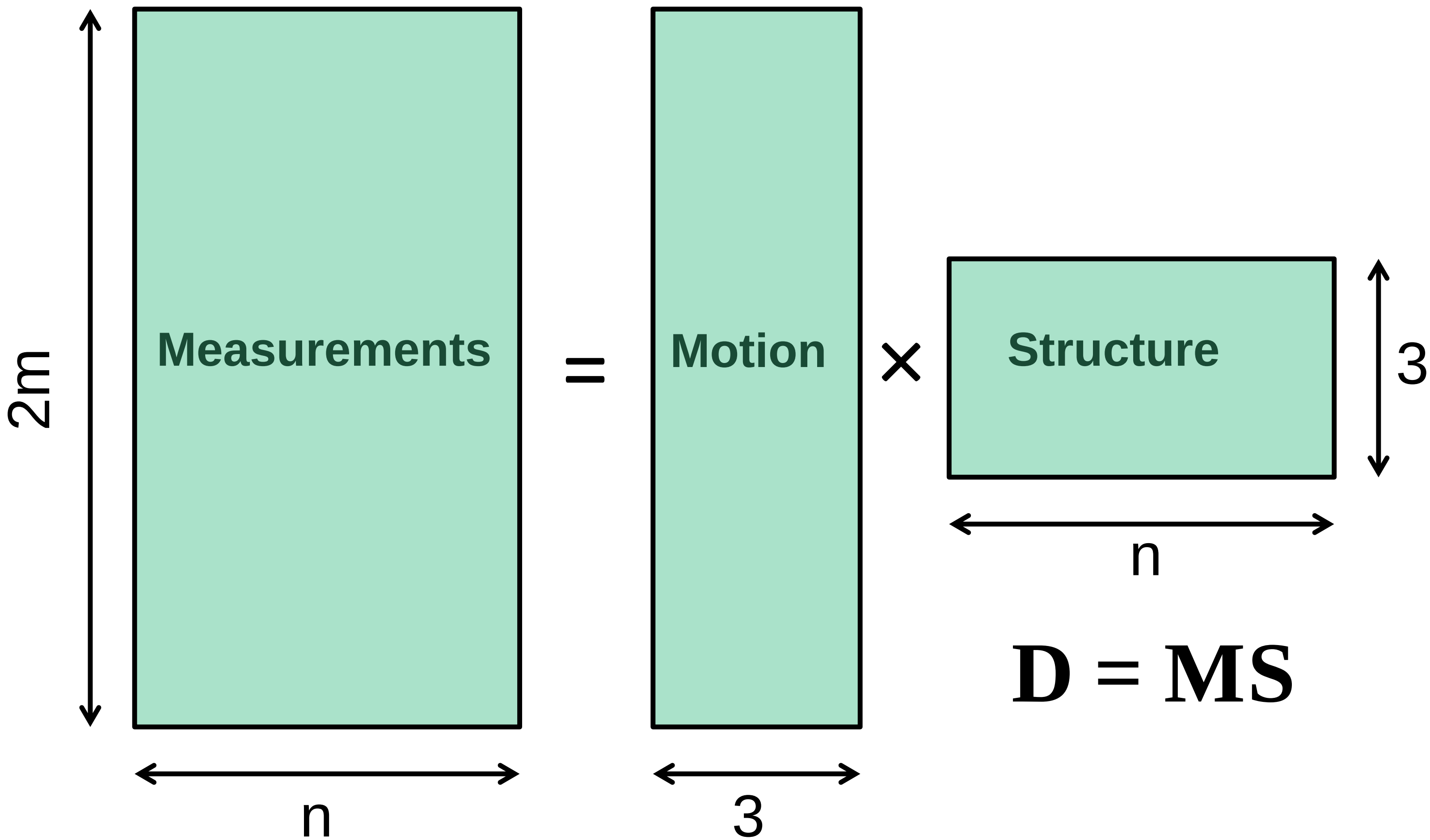
cameras (2*m* × 3)

points (3 × *n*)

The measurement matrix **D = M S** has rank 3

(it's a product of a 2mx3 matrix and 3xn matrix)

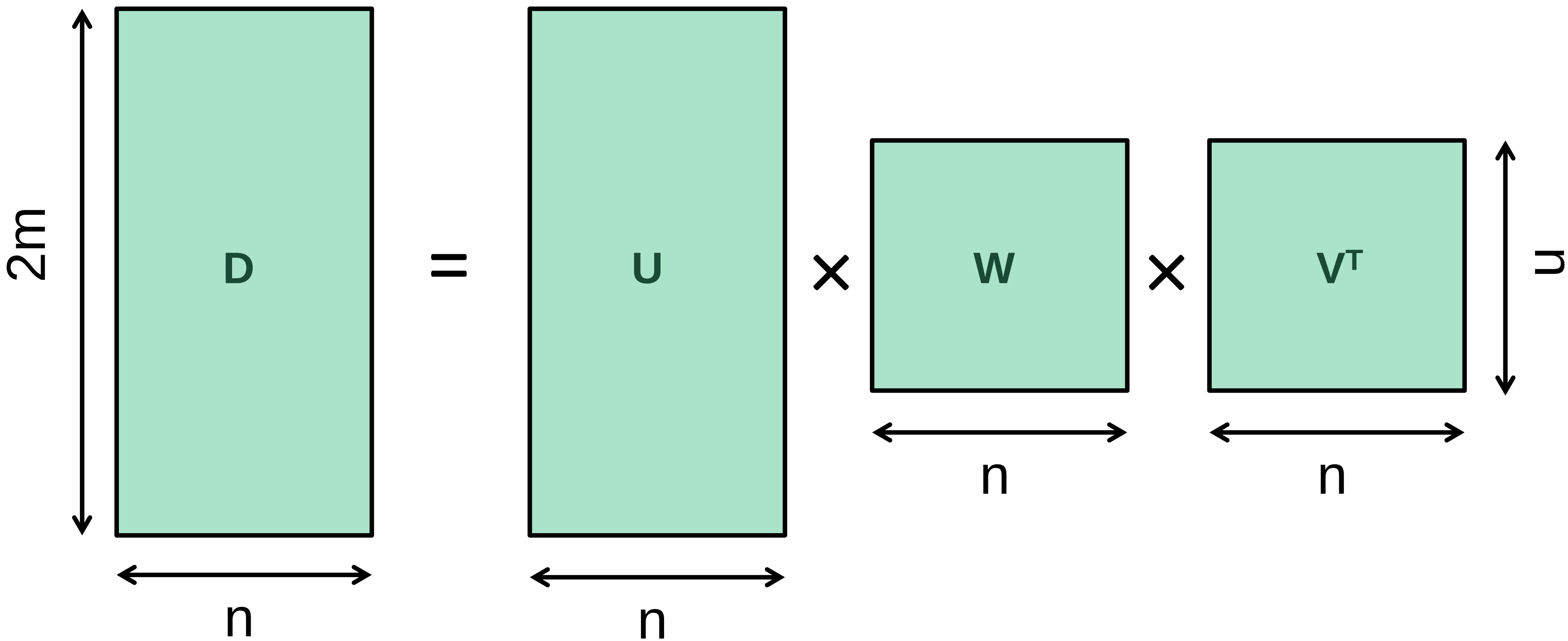
Factorizing the Measurement Matrix



$$\mathbf{D} = \mathbf{MS}$$

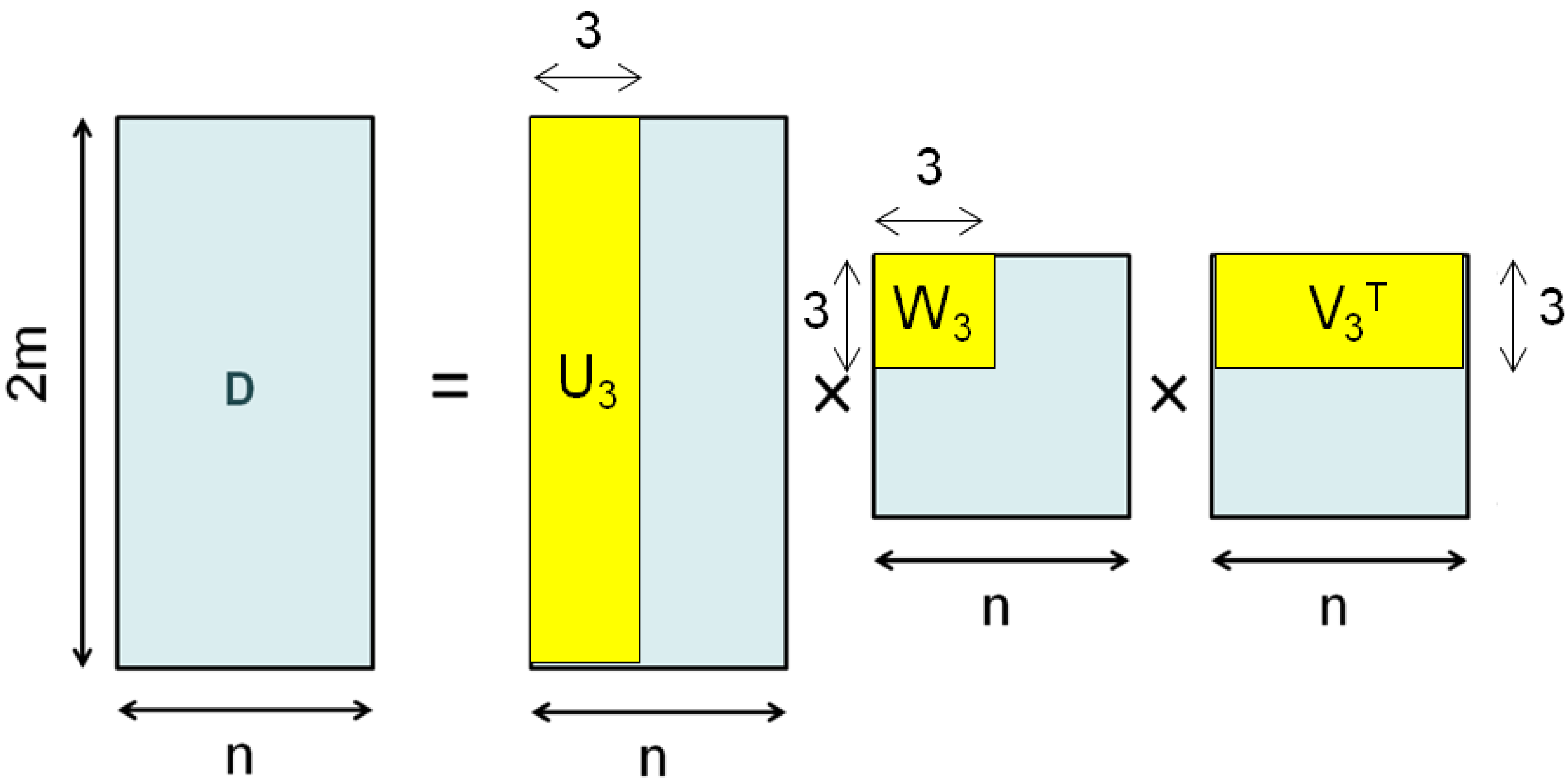
Factorizing the Measurement Matrix

- Singular value decomposition of D:

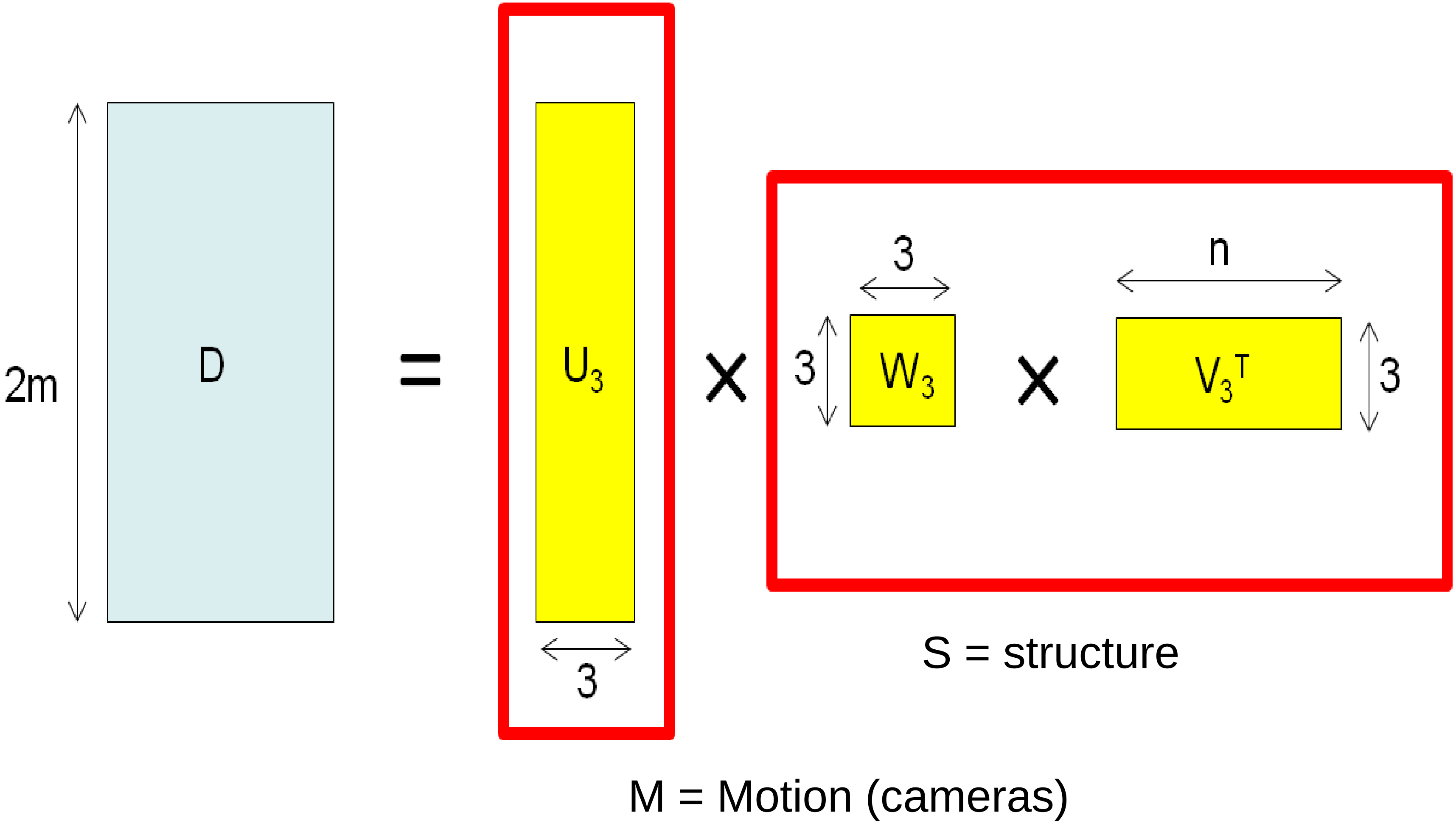


Factorizing the Measurement Matrix

Since rank (D)=3, there are only 3 non-zero singular values



Factorizing the Measurement Matrix

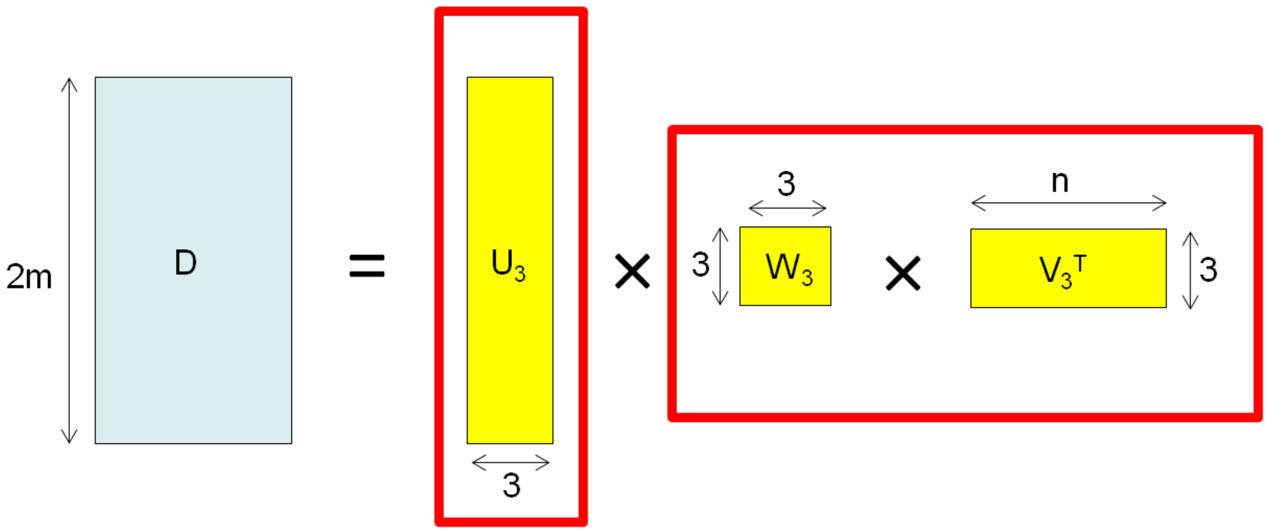


Factorizing the Measurement Matrix

What is the issue here?

D has rank>3 because of:

- measurement noise
- affine approximation



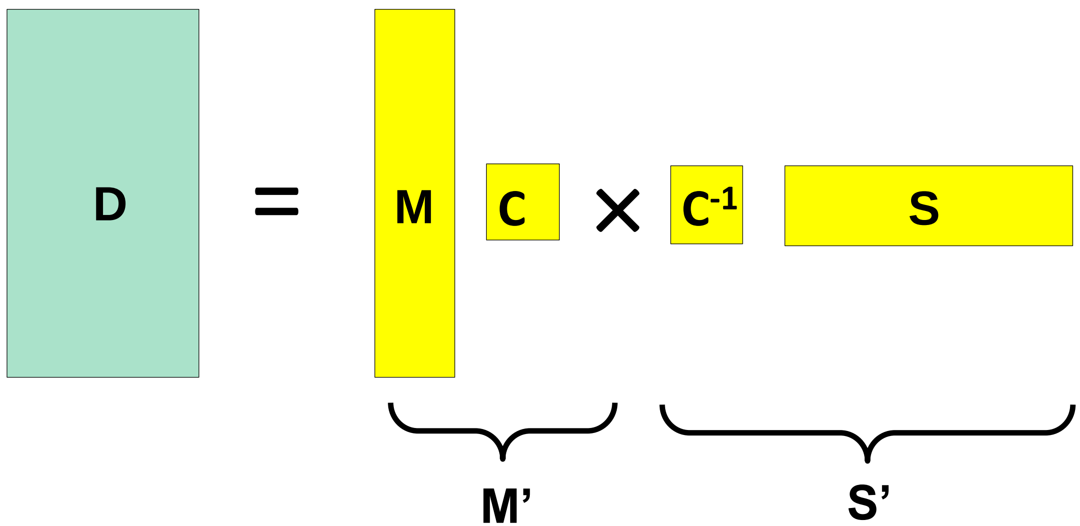
Theorem: When $\mathbf{D}$ has a rank greater than p, $\mathbf{U}_p\mathbf{W}_p\mathbf{V}_p^T$ is the best possible rank- p approximation of $\mathbf{A}$ in the sense of the Frobenius norm.

$$\mathbf{D} = \mathbf{U}_3\mathbf{W}_3\mathbf{V}_3^T$$

$$\begin{cases} \mathbf{A}_0 = \mathbf{U}_3 \\ \mathbf{P}_0 = \mathbf{W}_3\mathbf{V}_3^T \end{cases}$$

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} = \sqrt{\sum_{i=1}^{\min\{m,n\}}\sigma_i^2}$$

Affine Ambiguity



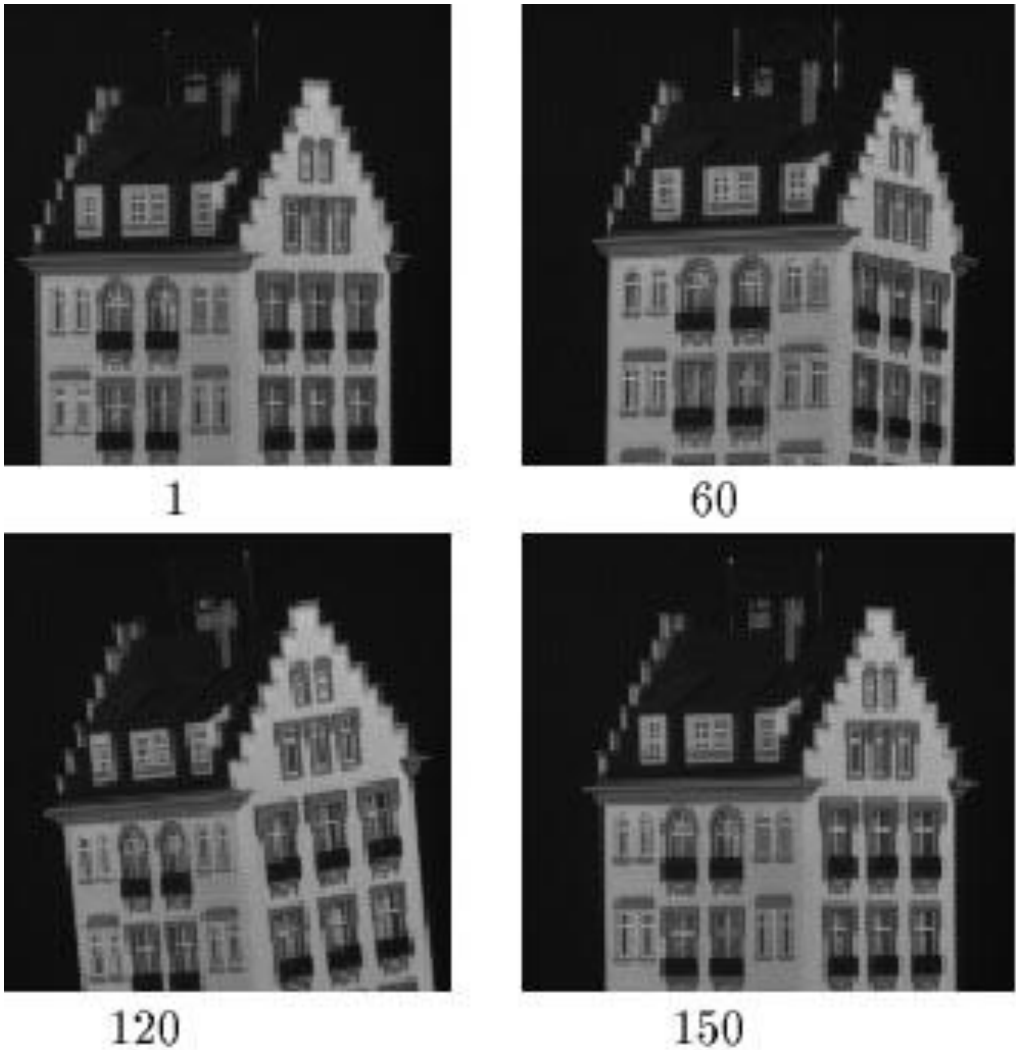
- The decomposition is not unique. We get the same **D** by using any 3×3 matrix **C** and applying the transformations:

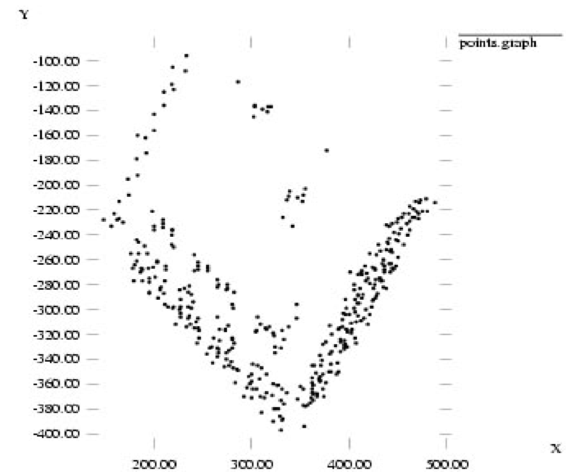
$$\mathbf{M} \rightarrow \mathbf{MC}$$

$$\mathbf{S} \rightarrow \mathbf{C^{-1}S}$$

- Additional constraints must be enforced to resolve this ambiguity

Reconstruction results





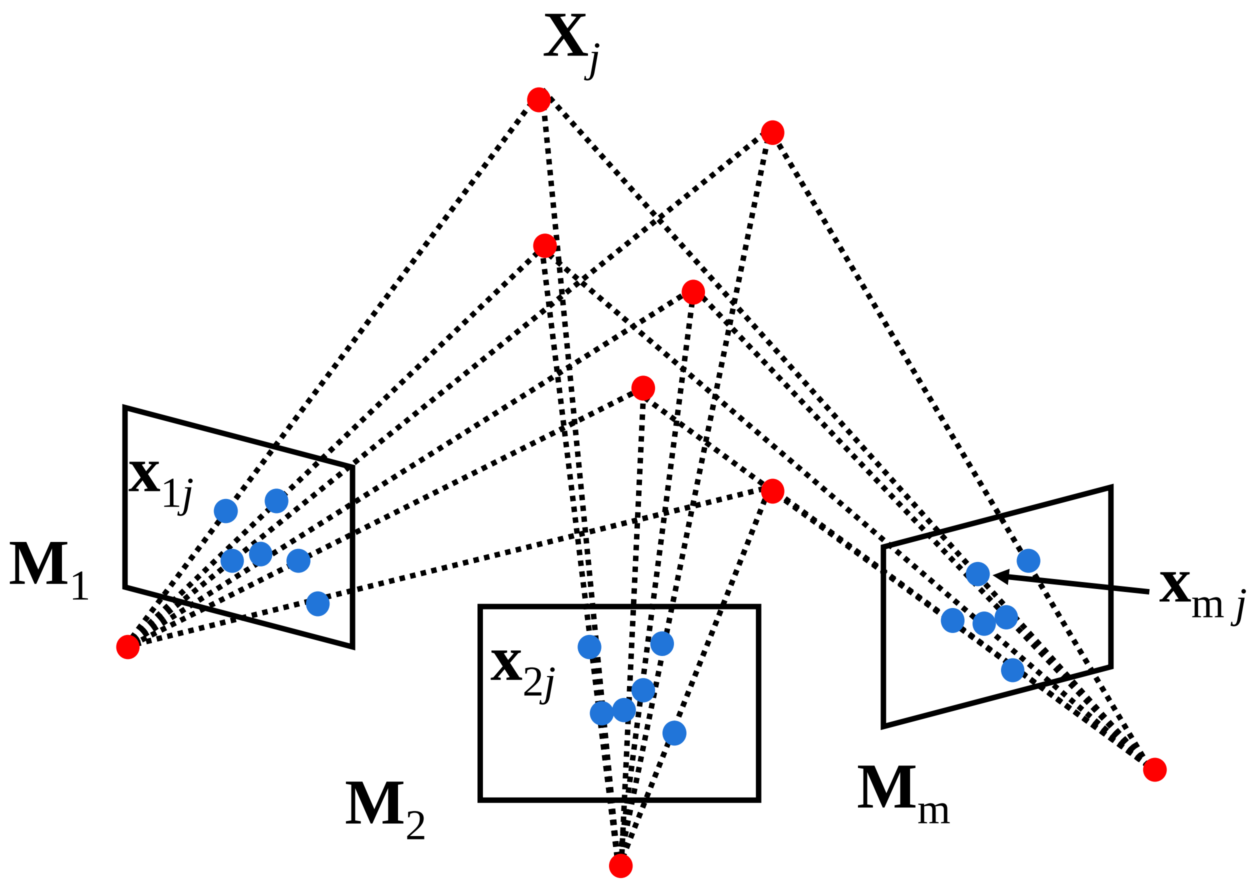
C. Tomasi and T. Kanade. [Shape and motion from image streams under orthography: A factorization method.](#) *IJCV*, 9(2):137-154, November 1992.

Lecture 6

Multi-view geometry

- The SFM problem
- Affine SFM
- Perspective SFM
- Self-calibration

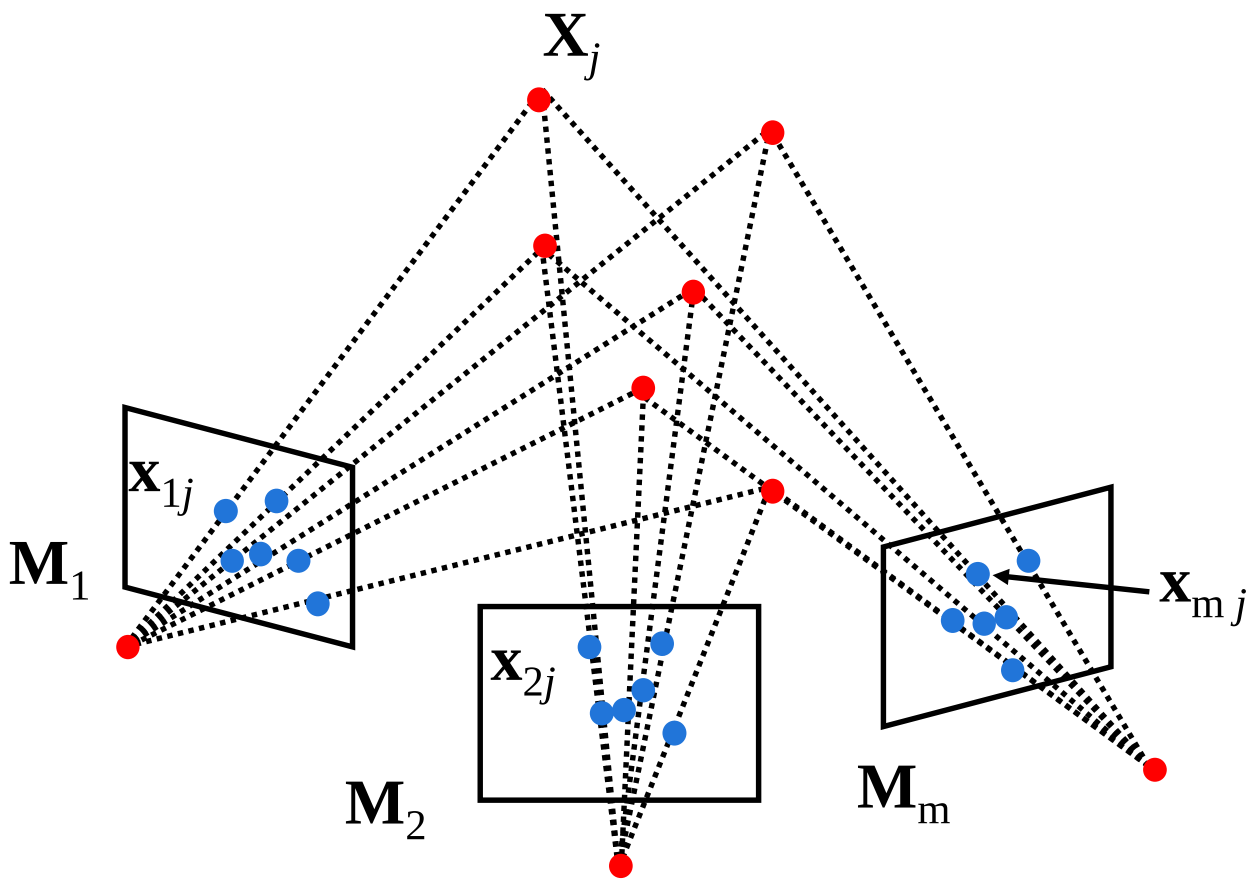
Structure from motion problem



From the mxn correspondences $\mathbf{x}_{ij}$, estimate:

- *m* projection matrices $\mathbf{M}_i$
- *n* 3D points $\mathbf{X}_j$
- Upgrade to metric reconstruction

Structure from motion problem



m cameras $M_1 \ldots M_m$

$$M_i = \begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & 1 \end{bmatrix}$$

The Structure-from-Motion Problem

Given *m* images of *n* fixed points X_j we can write

$$x_{ij} = M_i \; X_j \quad \text{for} \quad i = 1, \ldots, m \quad \text{and} \quad j = 1, \ldots, n.$$

Problem: estimate the m 3×4 matrices M_i and the n positions X_j from the m×n correspondences x_{ij} .

• With no calibration info, cameras and points can only be recovered up to a 4x4 projective (16 parameters)

• Given two cameras, how many points are needed?

• How many equations and how many unknown?

2m × n equations in 11m+3n – 16 unknowns

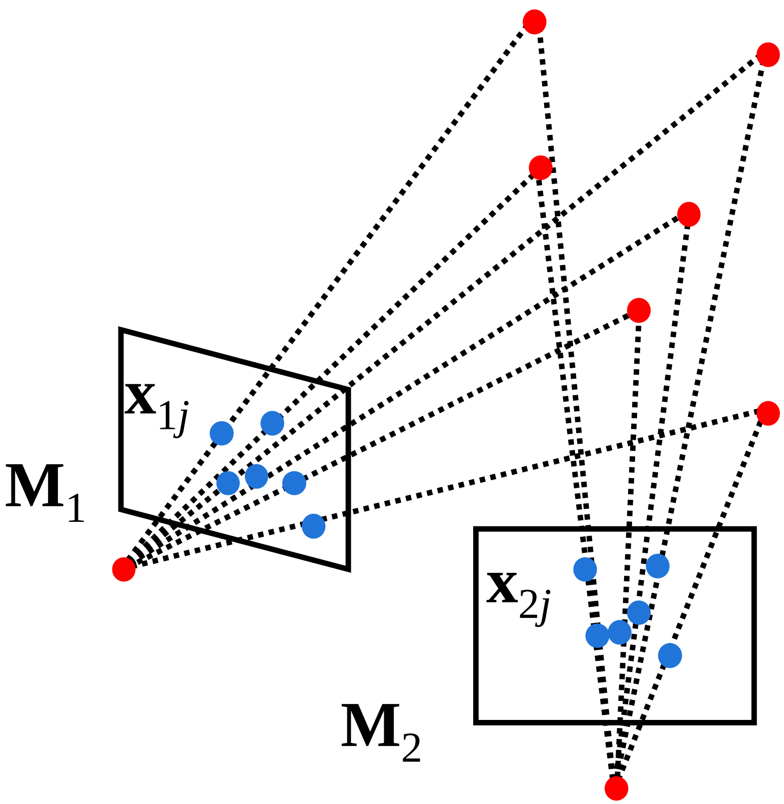
Structure-from-Motion Algorithms

- Algebraic approach (by fundamental matrix)
- Factorization method (by SVD)
- Bundle adjustment

Algebraic approach (2-view case)

1. Compute the fundamental matrix F from two views (eg. 8 point algorithm)
2. Use F to estimate projective cameras
3. Use these cameras to triangulate and estimate points in 3D

Algebraic approach (2-view case)



$$x_{ij} = M_i \, X_j$$

Apply a projective transformation H such that:

$$M_1 H^{-1} = [I \quad 0] \qquad\qquad M_2 H^{-1} = [A \quad b]$$

Canonical perspective cameras

Algebraic approach (2-view case)

$$\begin{cases} M_1 H^{-1} = [I \quad 0] \\ M_2 H^{-1} = [A \quad b] \\ \tilde{X} = H\,\mathbf{X} \end{cases}$$

$$\mathbf{x} = M_1 H^{-1} H\,\mathbf{X} = [\mathbf{I} \mid \mathbf{0}]\tilde{\mathbf{X}}$$

$$\mathbf{x}' = M_2 H^{-1} H\,\mathbf{X} = [\mathbf{A} \mid \mathbf{b}]\tilde{\mathbf{X}}$$

$$\mathbf{x}' = [\mathbf{A} \mid \mathbf{b}]\tilde{\mathbf{X}} = [\mathbf{A} \mid \mathbf{b}]\begin{bmatrix} \tilde{X}_1 \\ \tilde{X}_2 \\ \tilde{X}_3 \\ 1 \end{bmatrix} = \mathbf{A}[I \mid 0]\begin{bmatrix} \tilde{X}_1 \\ \tilde{X}_2 \\ \tilde{X}_3 \\ 1 \end{bmatrix} + \mathbf{b} = \mathbf{A}[I \mid 0]\,\tilde{\mathbf{X}} + b \overset{?}{=} A\mathbf{x} + b$$

$$\mathbf{x}' \times \mathbf{b} = (\mathbf{A}\mathbf{x} + \mathbf{b}) \times \mathbf{b} = \mathbf{A}\mathbf{x} \times \mathbf{b}$$

$$\mathbf{x}'^T \cdot (\mathbf{x}' \times \mathbf{b}) = \mathbf{x}'^T \cdot (\mathbf{A}\mathbf{x} \times \mathbf{b}) = 0$$

$$\mathbf{x}'^T (\mathbf{b} \times \mathbf{A}\mathbf{x}) = 0$$

Cross product as matrix multiplication

$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = [\mathbf{a}_\times]\mathbf{b}$$

Algebraic approach (2-view case)

$$\mathbf{x} = M_1\ H^{-1}\ H\ \mathbf{X} = [\mathbf{I}\,|\,\mathbf{0}]\tilde{\mathbf{X}}$$

$$\mathbf{x}' = M_2\ H^{-1}\ H\ \mathbf{X} = [\mathbf{A}\,|\,\mathbf{b}]\tilde{\mathbf{X}}$$

$$\begin{cases} \mathrm{M}_1\ \mathrm{H}^{-1} = \begin{bmatrix}\mathrm{I} & 0\end{bmatrix} \\ \mathrm{M}_2\ \mathrm{H}^{-1} = \begin{bmatrix}\mathrm{A} & \mathrm{b}\end{bmatrix} \\ \tilde{\mathbf{X}} = \mathrm{H}\ \mathbf{X} \end{cases}$$

$$\mathbf{x}' = [\mathbf{A}\,|\,\mathbf{b}]\tilde{\mathbf{X}} = [\mathbf{A}\,|\,\mathbf{b}]\begin{bmatrix}\tilde{\mathrm{X}}_1 \\ \tilde{\mathrm{X}}_2 \\ \tilde{\mathrm{X}}_3 \\ 1\end{bmatrix} = \mathbf{A}[\mathrm{I}\,|\,0]\begin{bmatrix}\tilde{\mathrm{X}}_1 \\ \tilde{\mathrm{X}}_2 \\ \tilde{\mathrm{X}}_3 \\ 1\end{bmatrix} + \mathbf{b} = \mathbf{A}[\mathrm{I}\,|\,0]\,\tilde{\mathbf{X}} + \mathrm{b} \overset{?}{=} \mathrm{A}\mathbf{x} + \mathrm{b}$$

$$\mathbf{x}' \times \mathbf{b} = (\mathbf{A}\mathbf{x} + \mathbf{b}) \times \mathbf{b} = \mathbf{A}\mathbf{x} \times \mathbf{b}$$

$$\mathbf{x}'^T \cdot (\mathbf{x}' \times \mathbf{b}) = \mathbf{x}'^T \cdot (\mathbf{A}\mathbf{x} \times \mathbf{b}) = 0$$

$$\mathbf{x}'^T (\mathbf{b} \times \mathbf{A}\mathbf{x}) = 0$$

$$\mathbf{x}'^{\mathrm{T}} [\mathbf{b}_\times]\mathbf{A}\,\mathbf{x} = 0$$

is this familiar?

$$\mathbf{F} = [\mathbf{b}_\times]\mathbf{A}$$

$$\mathbf{x}'^{\mathrm{T}} \mathrm{F}\,\mathbf{x} = 0$$

Compute cameras

$$\mathbf{x}'^{T} F \mathbf{x} = 0 \qquad \mathbf{F} = [\mathbf{b}_{\times}]\mathbf{A}$$

Compute **b**:

$$\mathbf{F} \cdot \mathbf{b} = [\mathbf{b}_{\times}]\mathbf{A} \cdot \mathbf{b} = \mathbf{b} \times \mathbf{A} \cdot \mathbf{b} = 0 \rightarrow$$

- **F** is singular
- Compute **b** as least sq. solution of **F b** = 0, with |**b**|=1 using SVD

Compute **A**: $\mathbf{A} = \mathbf{A}' = -[\mathbf{b}_{\times}]\,\mathbf{F}$

Indeed, let's verify that $[\mathbf{b}_{\times}]\mathbf{A}'$ is still equal to $\mathbf{F}$

Indeed: $[\mathbf{b}_{\times}]\mathbf{A}' = -[\mathbf{b}_{\times}][\mathbf{b}_{\times}]\,\mathbf{F} = (\mathbf{b}\,\mathbf{b}^{T} - |\mathbf{b}|^{2}\,\mathbf{I})\,\mathbf{F} = \mathbf{b}\,\mathbf{b}^{T}\mathbf{F} + |\mathbf{b}|^{2}\,\mathbf{F} = 0 + 1 \cdot \mathbf{F} = \mathbf{F}$

$$M^{p}_{1} = \begin{bmatrix} I & 0 \end{bmatrix} \qquad M^{p}_{2} = \begin{bmatrix} -[\mathbf{b}_{x}]\mathbf{F} & \mathbf{b} \end{bmatrix}$$

Algebraic approach (2-view case)

1. Compute the fundamental matrix F from two views (eg. 8 point algorithm)
2. Compute b and A from F
3. Use b and A to estimate projective cameras
4. Use these cameras to triangulate and estimate points in 3D

Interpretation of **b**

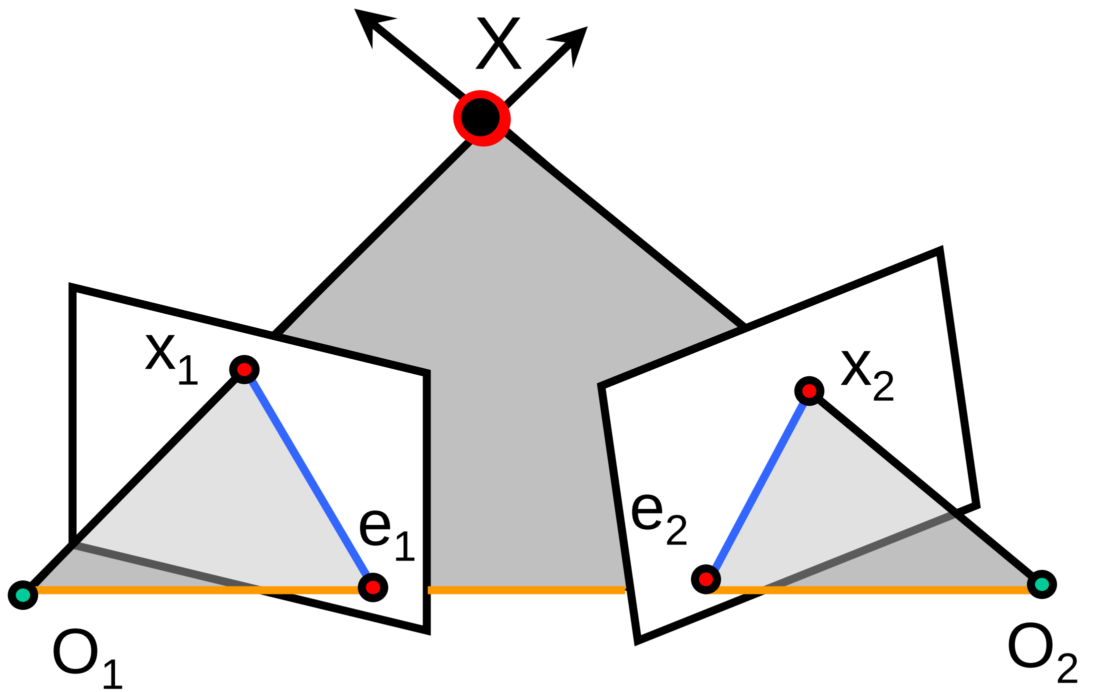
$$\mathbf{x}'^{\mathrm{T}} \mathrm{F}\, \mathbf{x} = 0 \qquad \mathbf{F} = [\mathbf{b}_{\times}]\mathbf{A}$$

$$\mathbf{F} \cdot \mathbf{b} = 0$$

$$\mathbf{A} = -[\mathbf{b}_{\times}]\, \mathbf{F}$$

What’s **b**??

Epipolar Constraint [lecture 6]



$F\,x_2$ is the epipolar line associated with x_2 ($l_1 = F\,x_2$)

$F^T x_1$ is the epipolar line associated with x_1 ($l_2 = F^T x_1$)

F is singular (rank two)

$F\,e_2 = 0$ and $F^T e_1 = 0$

F is 3x3 matrix; 7 DOF

Interpretation of **b**

$$\mathbf{x'}^{\mathrm{T}} \mathrm{F}\, \mathbf{x} = 0 \qquad \mathbf{F} = [\mathbf{b}_{\times}]\mathbf{A}$$

$$\mathbf{F} \cdot \mathbf{b} = 0$$

$$\mathbf{A} = -[\mathbf{b}_{\times}]\, \mathbf{F}$$

b is an epipole!

$$\mathrm{M}^{\mathrm{p}}{}_{1} = \begin{bmatrix} \mathrm{I} & 0 \end{bmatrix} \qquad M^{p}_{2} = \begin{bmatrix} -[\mathbf{e}_{x}]\mathbf{F} & \mathbf{e} \end{bmatrix}$$

HZ, page 254
PF, page 288

Structure-from-Motion Algorithms

- Algebraic approach (by fundamental matrix)
- Factorization method (by SVD)
- Bundle adjustment

Limitations of the approaches seen so far

- Factorization methods assume all points are visible.

This not true if:

  - occlusions occur
  - failure in establishing correspondences

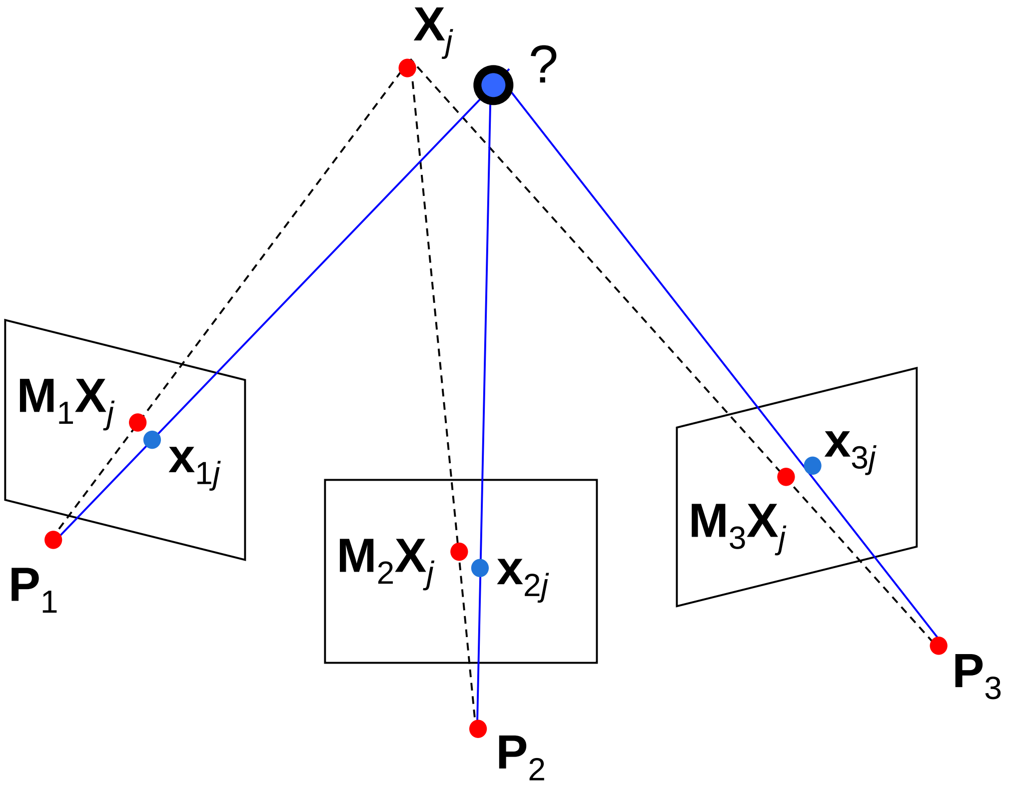
- Algebraic methods work with 2 views

Bundle adjustment

Non-linear method for refining structure and motion

Minimizing re-projection error

$$E(\mathrm{M}, \mathbf{X}) = \sum_{i=1}^{m} \sum_{j=1}^{n} D\left(\mathbf{x}_{ij}, \mathrm{M}_i \mathbf{X}_j\right)^2$$



Bundle adjustment

Non-linear method for refining structure and motion

Minimizing re-projection error

$$E(M, \mathbf{X}) = \sum_{i=1}^{m} \sum_{j=1}^{n} D\left(\mathbf{x}_{ij}, M_i \mathbf{X}_j\right)^2$$

- Advantages
  - Handle large number of views
  - Handle missing data

- Limitations
  - Large minimization problem (parameters grow with number of views)
  - requires good initial condition

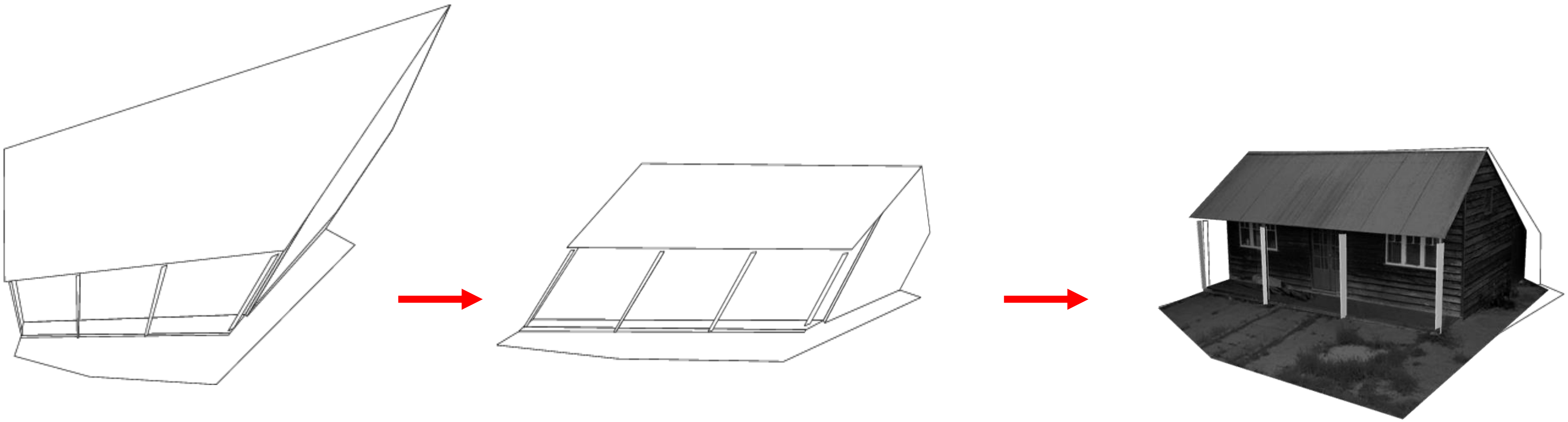
→ Used as the final step of SFM

Lecture 6
Multi-view geometry

- The SFM problem
- Affine SFM
- Perspective SFM
- Self-calibration

Reading: [HZ] Chapters: 10,18,19
[FP] Chapter: 13

Recovering the metric reconstruction from the perspective (or affine) one is called self-calibration



SFM problem - summary

1. Estimate structure and motion up perspective transformation
    1. Algebraic
    2. factorization method
    3. bundle adjustment

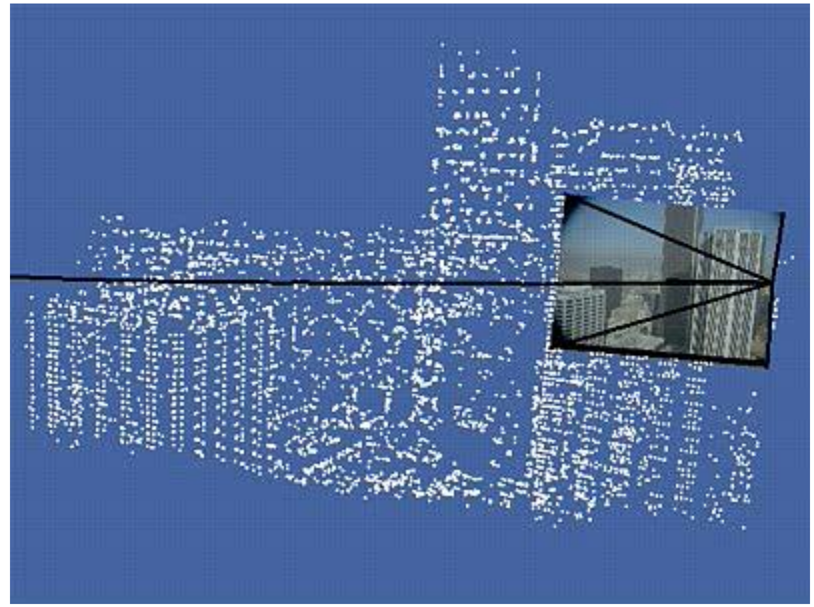
2. Convert from perspective to metric (self-calibration)

3. Bundle adjustment

** or **

1. Bundle adjustment with self-calibration constraints

Results and applications



Courtesy of Oxford **Visual Geometry Group**

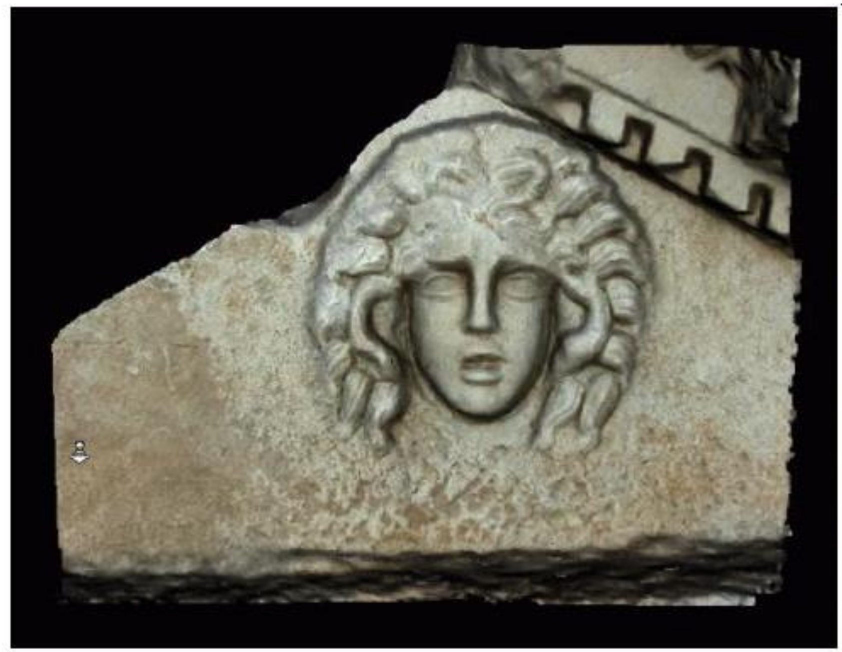
Lucas & Kanade, 81
Chen & Medioni, 92
Debevec et al., 96
Levoy & Hanrahan, 96
Fitzgibbon & Zisserman, 98
Triggs et al., 99
Pollefeys et al., 99
Kutulakos & Seitz, 99

Levoy et al., 00
Hartley & Zisserman, 00
Dellaert et al., 00
Rusinkiewic et al., 02
Nistér, 04
Brown & Lowe, 04
Schindler et al, 04
Lourakis & Argyros, 04
Colombo et al. 05

Golparvar-Fard, et al. JAEI 10
Pandey et al. IFAC , 2010
Pandey et al. ICRA 2011
Microsoft's PhotoSynth
Snavely et al., 06-08
Schindler et al., 08
Agarwal et al., 09
Frahm et al., 10

Results and applications

M. Pollefeys et al 98---



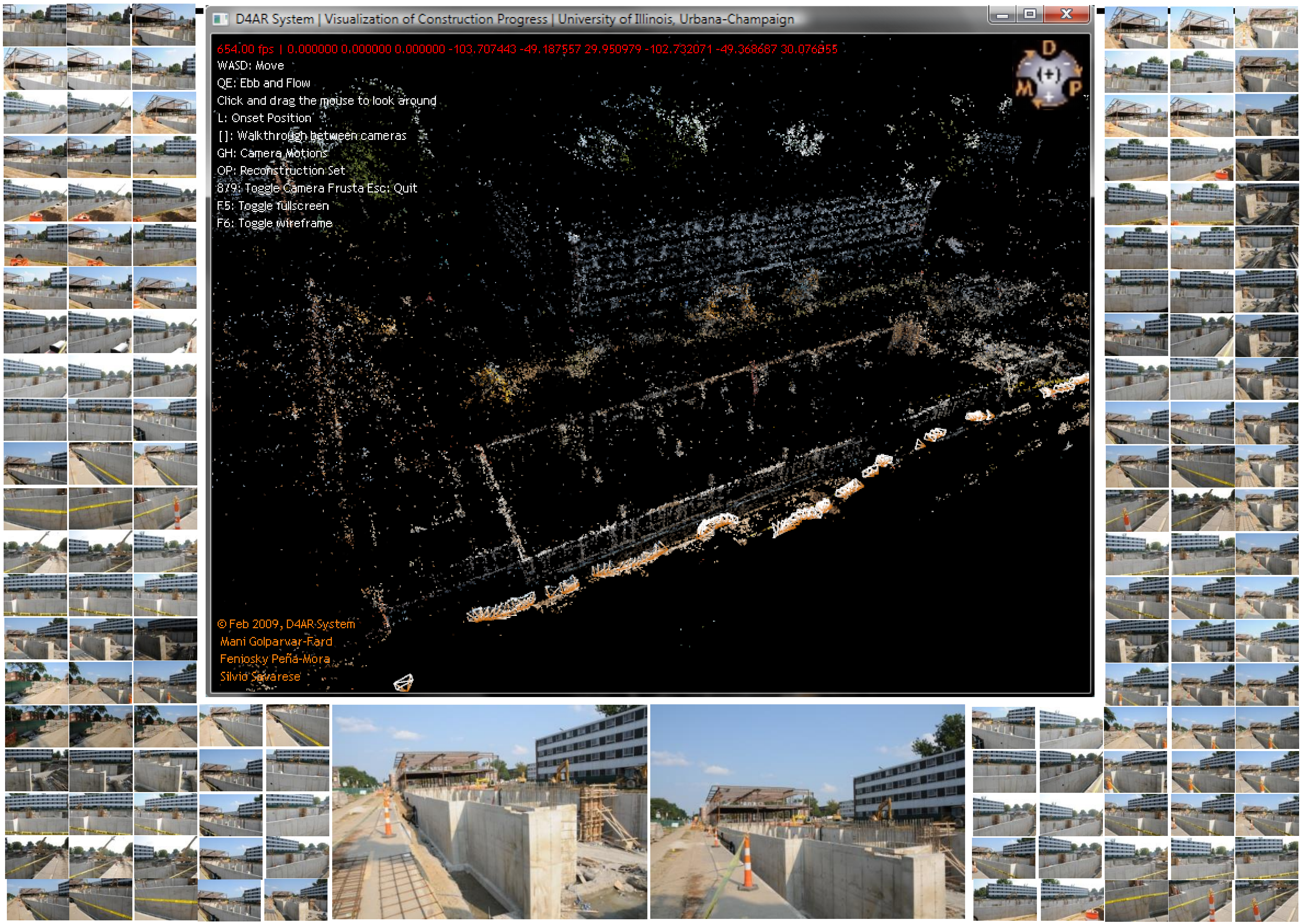
Results and applications

Noah Snavely, Steven M. Seitz, Richard Szeliski, "Photo tourism: Exploring photo collections in 3D," ACM Transactions on Graphics (SIGGRAPH Proceedings),2006,

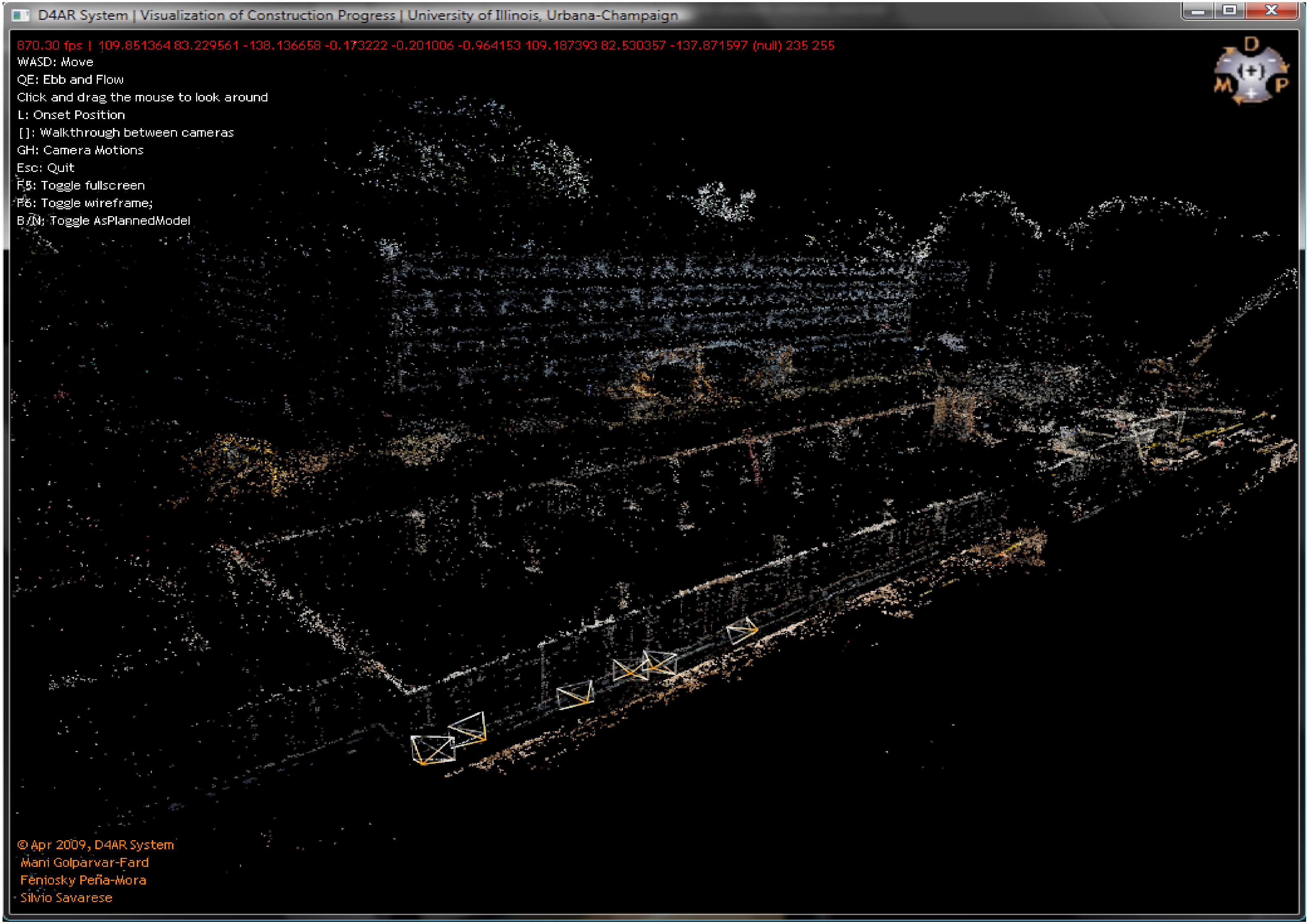
Microsoft Photosynth

Incremental reconstruction of construction sites

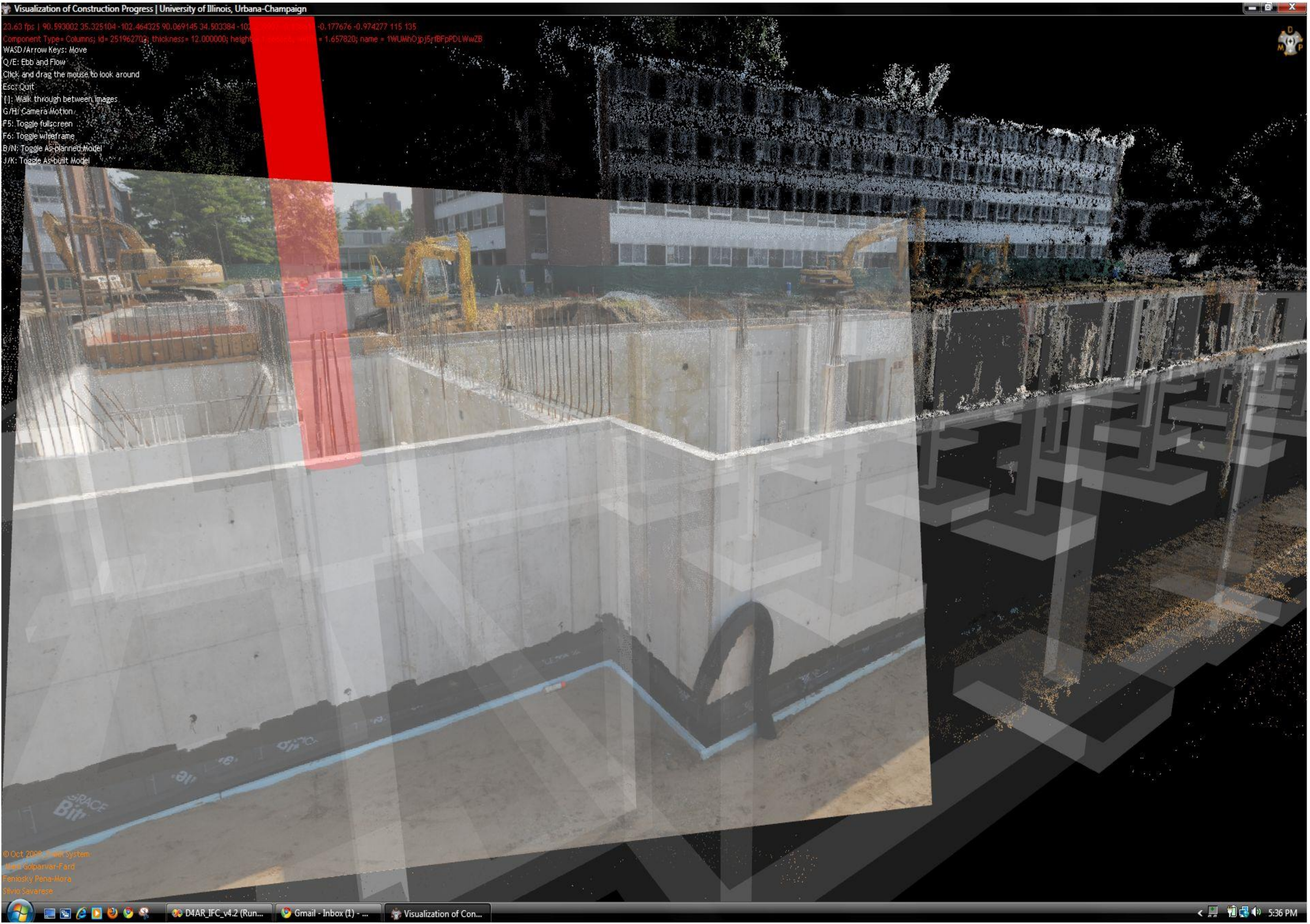
Initial pair – 2168 & Complete Set 62,323 points, 160 images



Reconstructed scene + Site photos



Reconstructed scene + Site photos



Next lecture

Self-calibration

Volumetric stereo