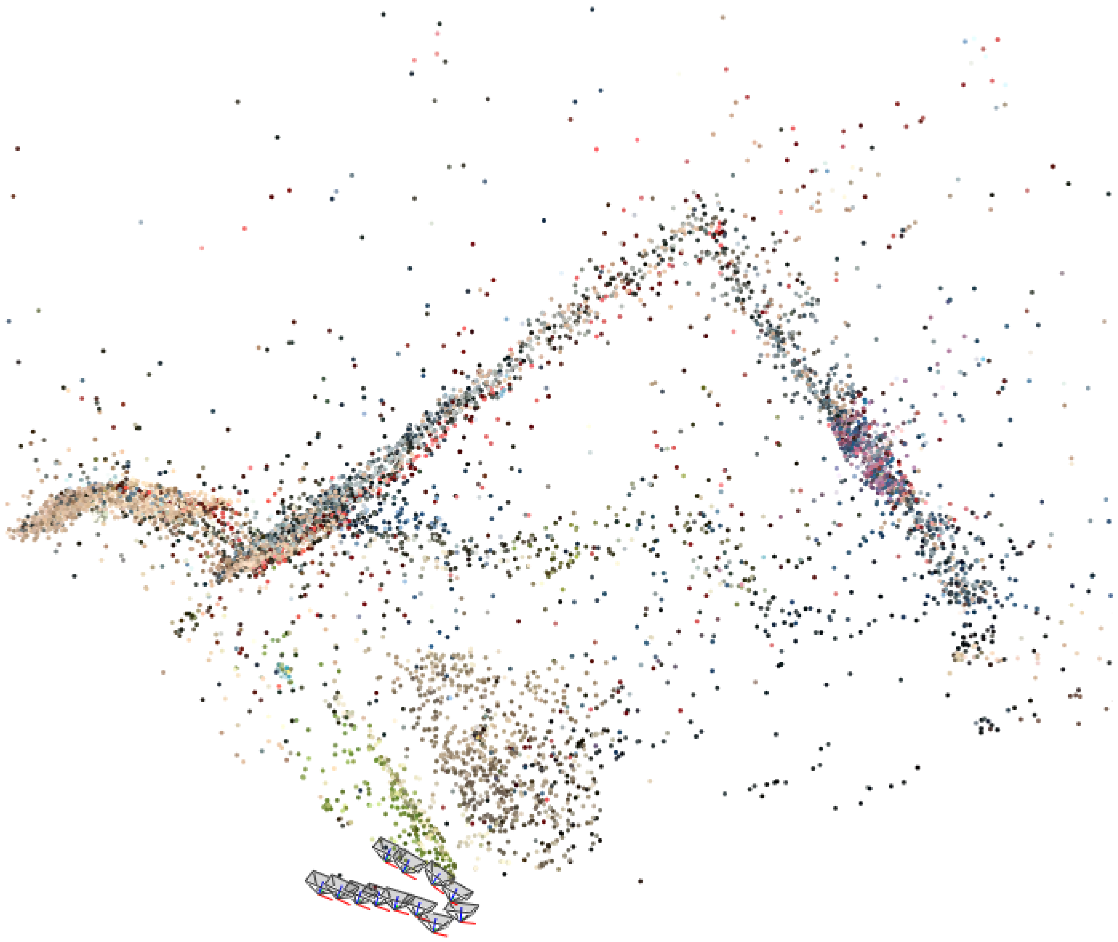


Bundle Adjustment

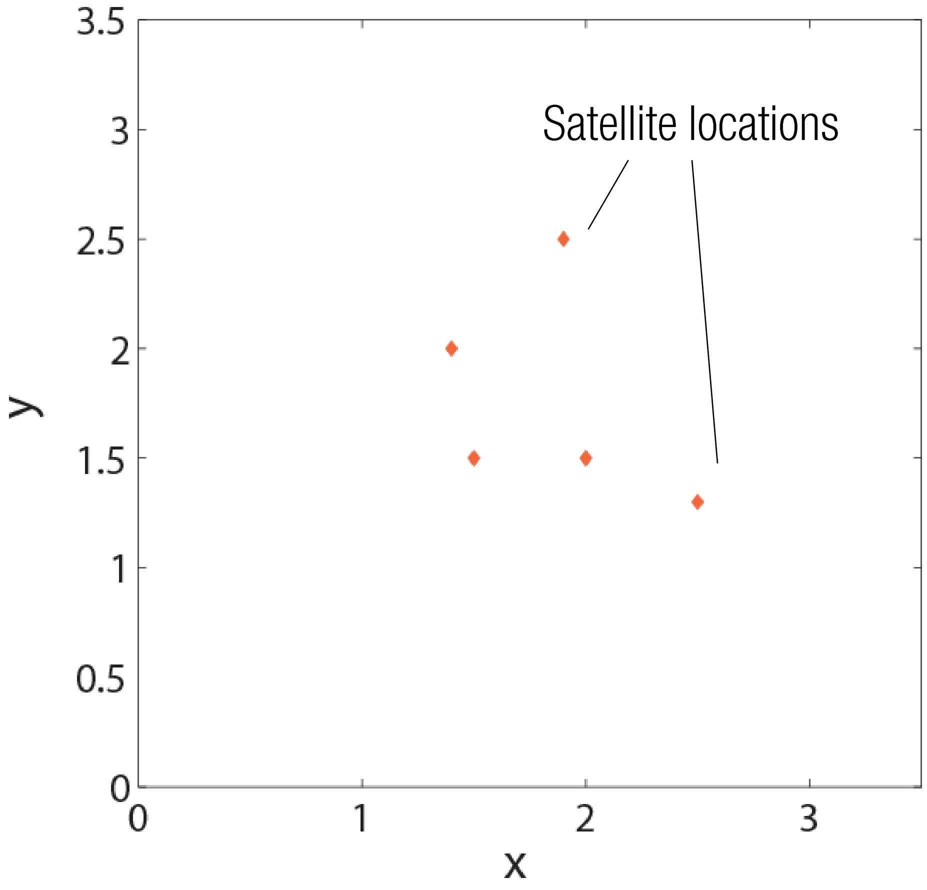
Announcement

- HW #6 is out
  - Two deadlines: May 3 (Problem 2,3,4,5) and May 10 (Problem 6 with full reconstruction)
  - 5980 students: < 5 image reconstruction
  - 8980 students: <10 image reconstruction

- Jiawei Mo will present a paper on Tuesday

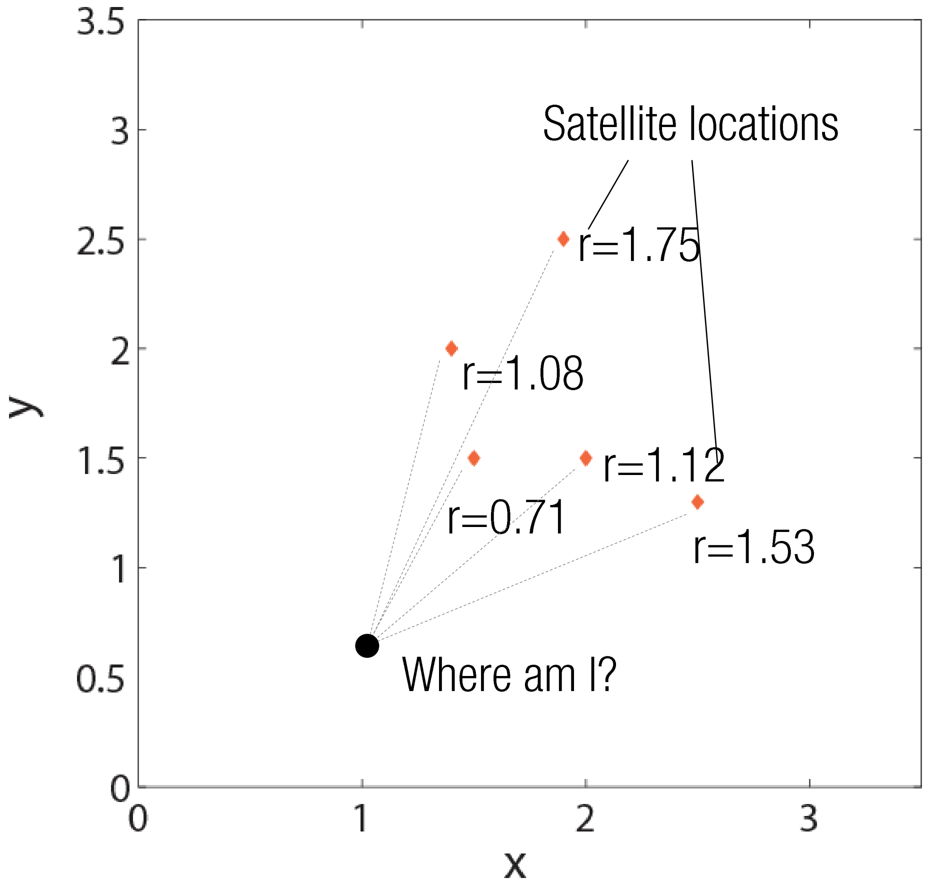
Where am I?: GPS Localization

Example:
Localization using range data from beacons



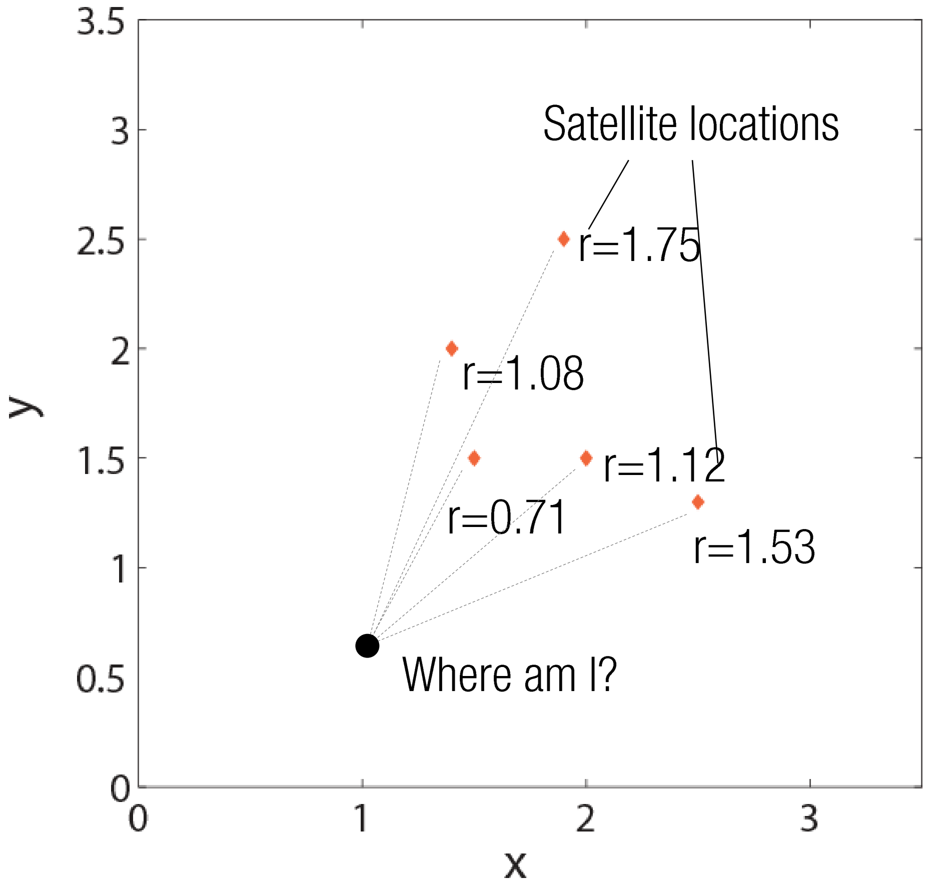
Where am I?: GPS Localization

Example:
Localization using range data from beacons



Where am I?: GPS Localization

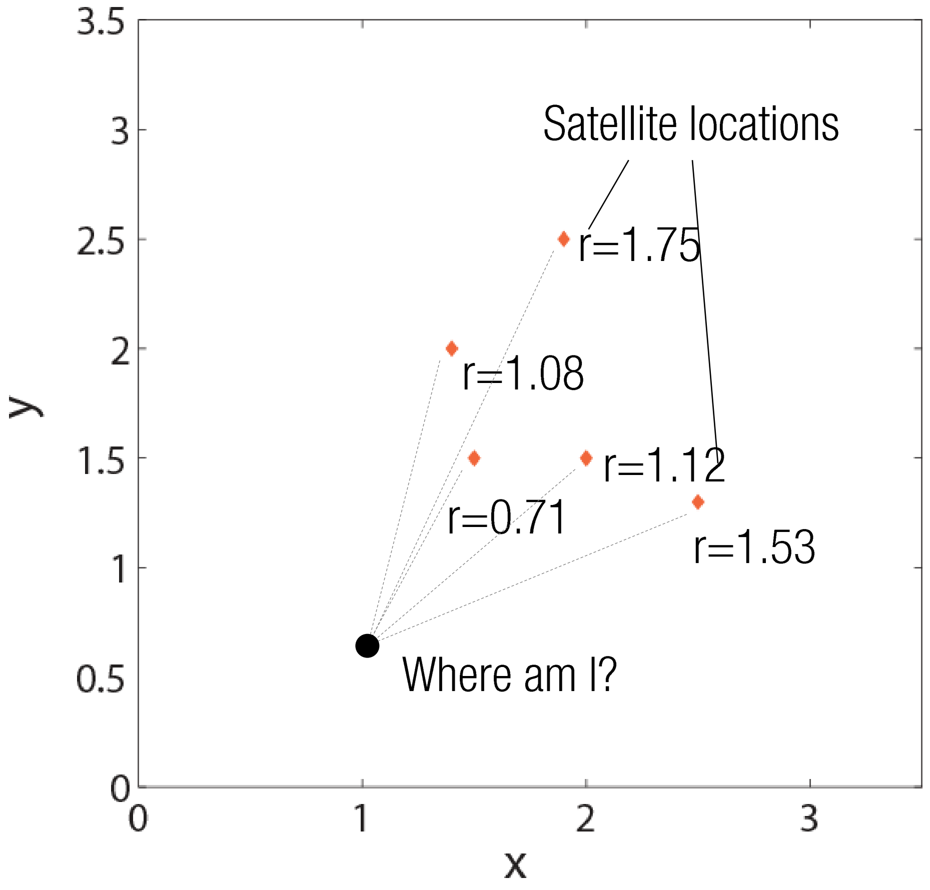
Example:
Localization using range data from beacons



$E =$

Where am I?: GPS Localization

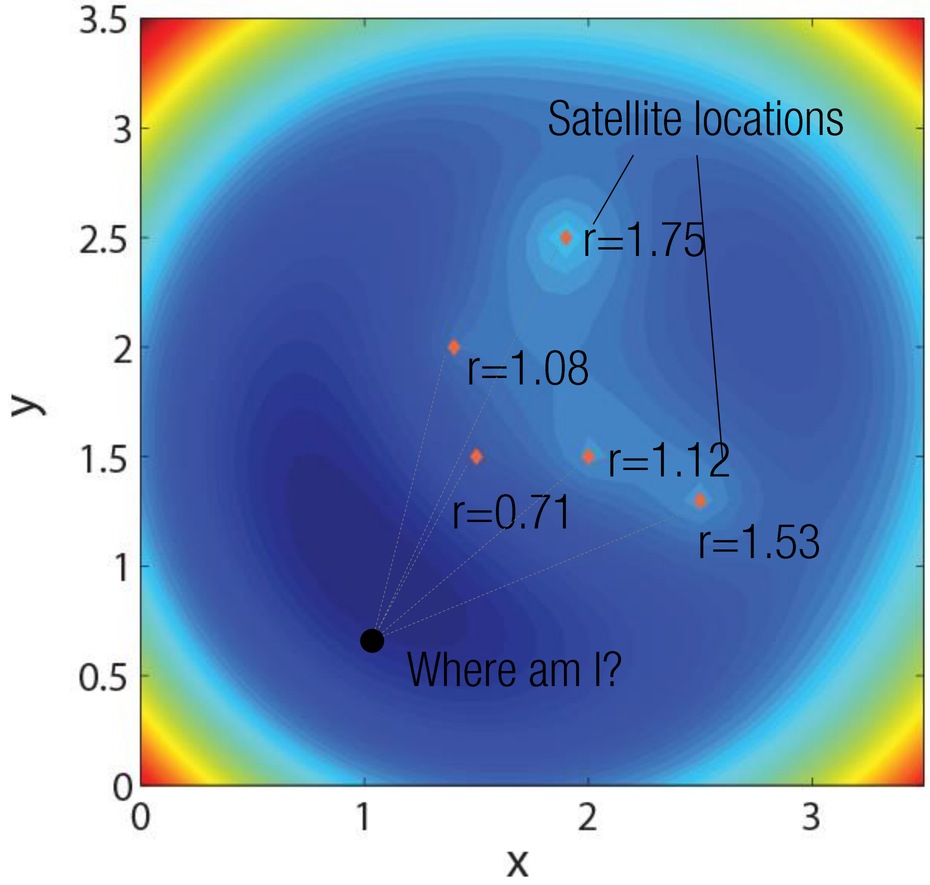
Example:
Localization using range data from beacons



$$E = \left\| \begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix} - \begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix} \right\|^2$$

Where am I?: GPS Localization

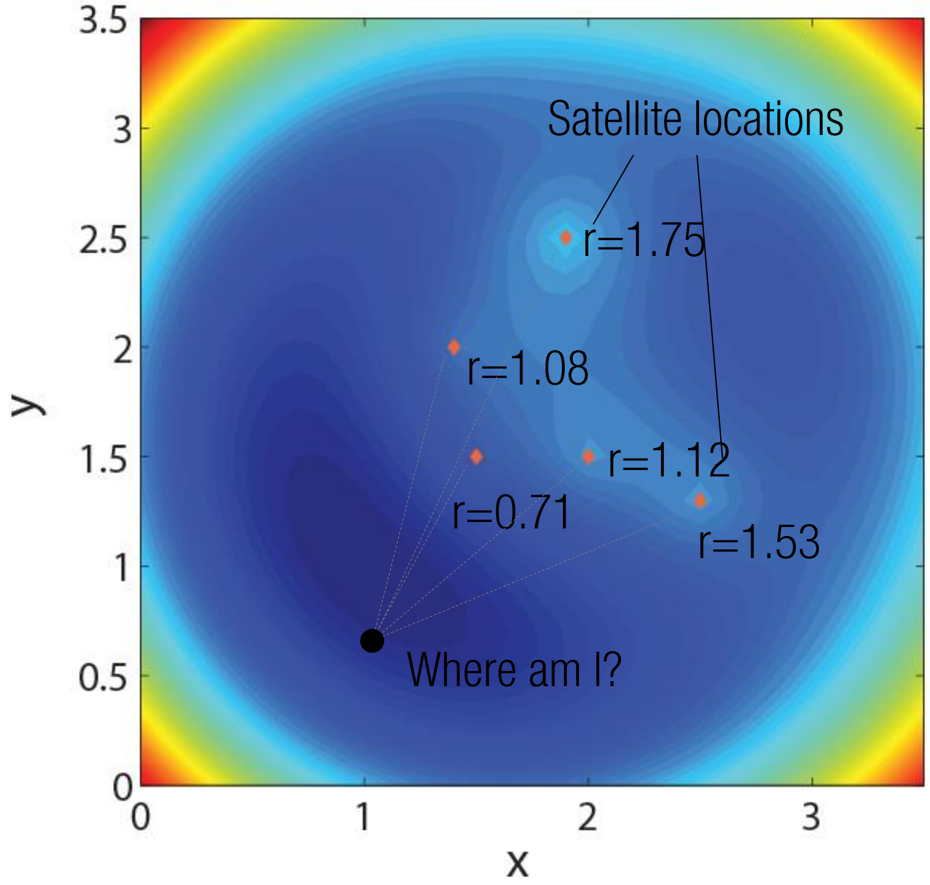
Example:
Localization using range data from beacons



$$E = \left\| \underbrace{\begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix}}_{f(\mathbf{x})} - \underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix}}_{\mathbf{b}} \right\|^2$$

Where am I?: GPS Localization

Example:
Localization using range data from beacons



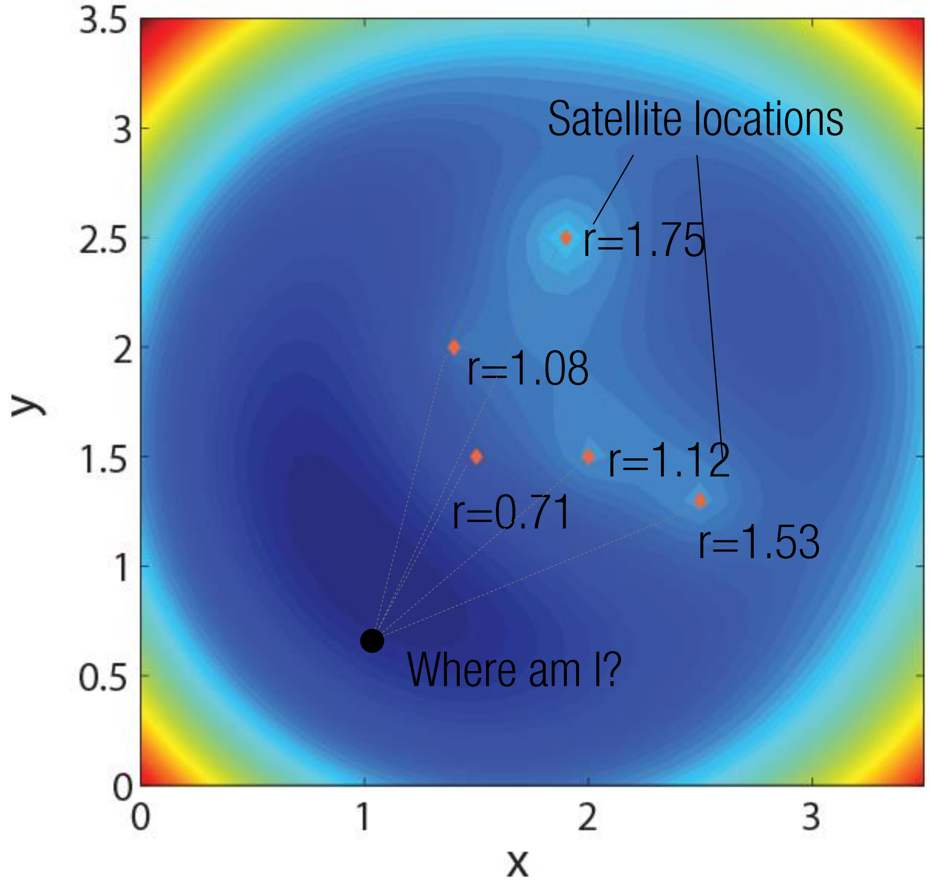
$$E = \left\| \underbrace{\begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix}}_{f(\mathbf{x})} - \underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix}}_{\mathbf{b}} \right\|^2$$

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Delta \mathbf{x} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

where $\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial \mathbf{x}_1} & \cdots & \frac{\partial f_1}{\partial \mathbf{x}_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_1}{\partial \mathbf{x}_n} & \cdots & \frac{\partial f_m}{\partial \mathbf{x}_n} \end{bmatrix}$: Jacobian

Where am I?: GPS Localization

Example:
Localization using range data from beacons



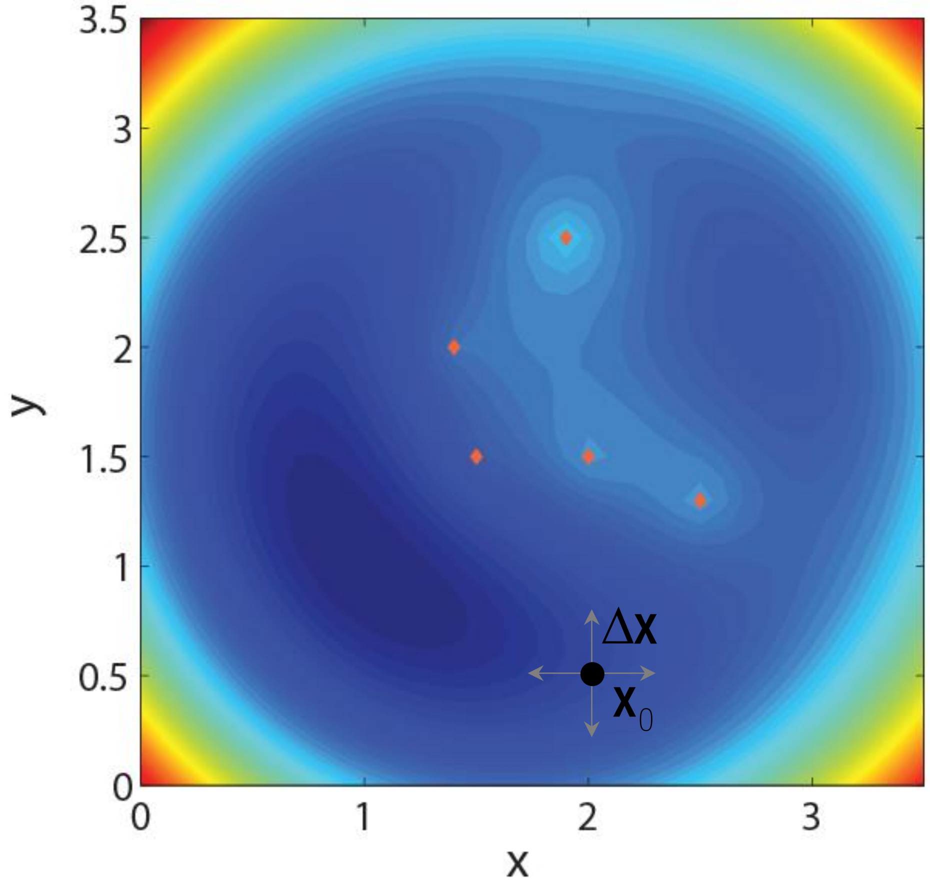
$$E = \left\| \underbrace{\begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix}}_{f(\mathbf{x})} - \underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix}}_{\mathbf{b}} \right\|^2$$

$$\frac{\partial f(\mathbf{x})^T}{\partial \mathbf{x}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Delta \mathbf{x} = \frac{\partial f(\mathbf{x})^T}{\partial \mathbf{x}} \left(\mathbf{b} - f(\mathbf{x}) \right)$$

$$\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} -\dfrac{u_1 - x}{\sqrt{(u_1 - x)^2 + (v_1 - y)^2}} & -\dfrac{v_1 - y}{\sqrt{(u_1 - x)^2 + (v_1 - y)^2}} \\ \vdots & \vdots \\ -\dfrac{u_5 - x}{\sqrt{(u_5 - x)^2 + (v_5 - y)^2}} & -\dfrac{v_5 - y}{\sqrt{(u_5 - x)^2 + (v_5 - y)^2}} \end{bmatrix}$$

Where am I?: GPS Localization

Example:
Localization using range data from beacons



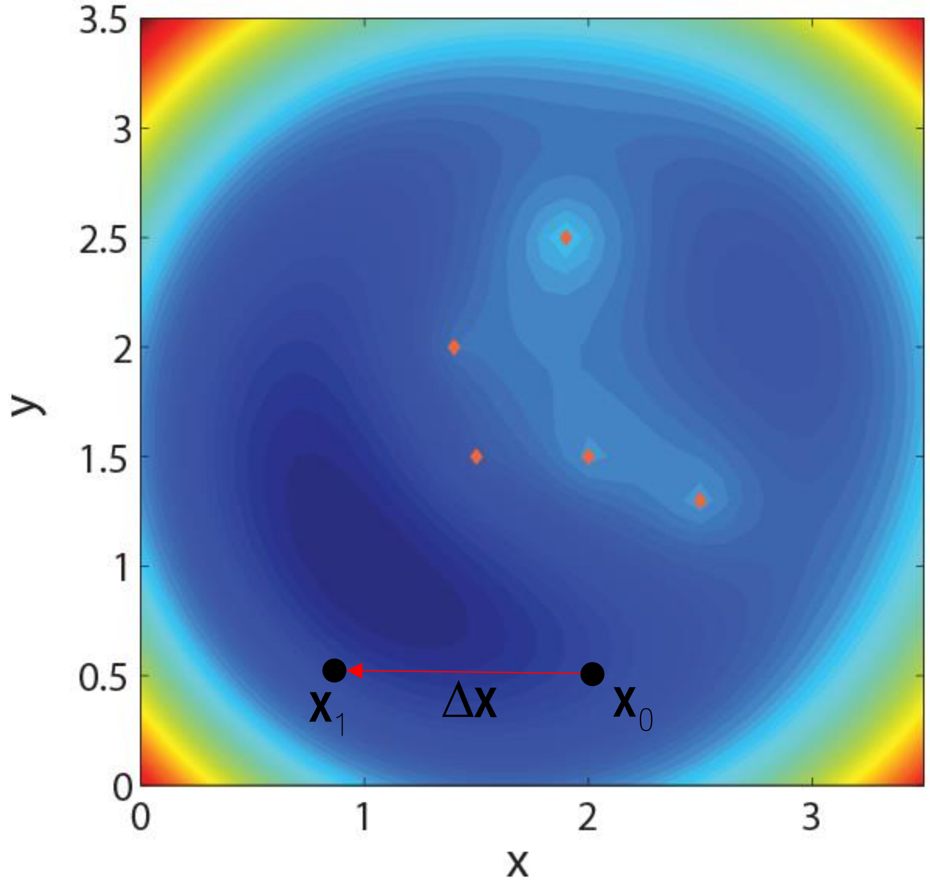
$$E = \left\| \underbrace{\begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix}}_{f(\mathbf{x})} - \underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix}}_{\mathbf{b}} \right\|^2$$

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Delta\mathbf{x} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

$$\Delta\mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

Where am I?: GPS Localization

Example:
Localization using range data from beacons



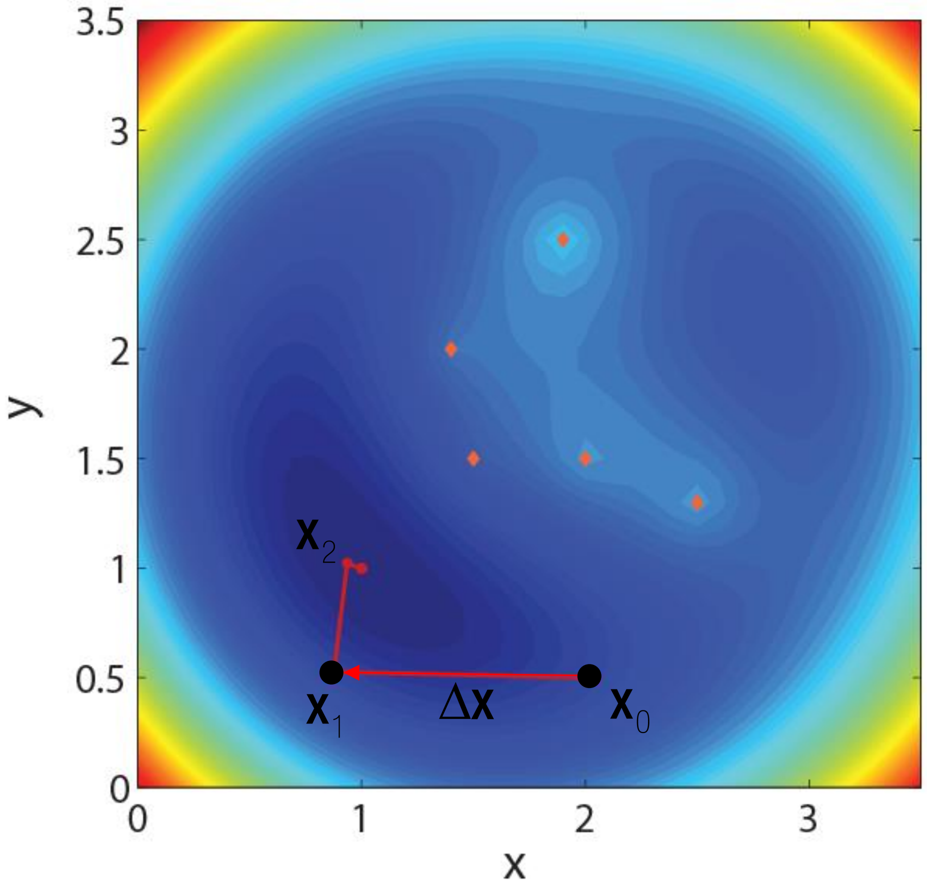
$$E = \left\| \underbrace{\begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix}}_{f(\mathbf{x})} - \underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix}}_{\mathbf{b}} \right\|^2$$

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Delta\mathbf{x} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

$$\Delta\mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

Where am I?: GPS Localization

Example:
Localization using range data from beacons



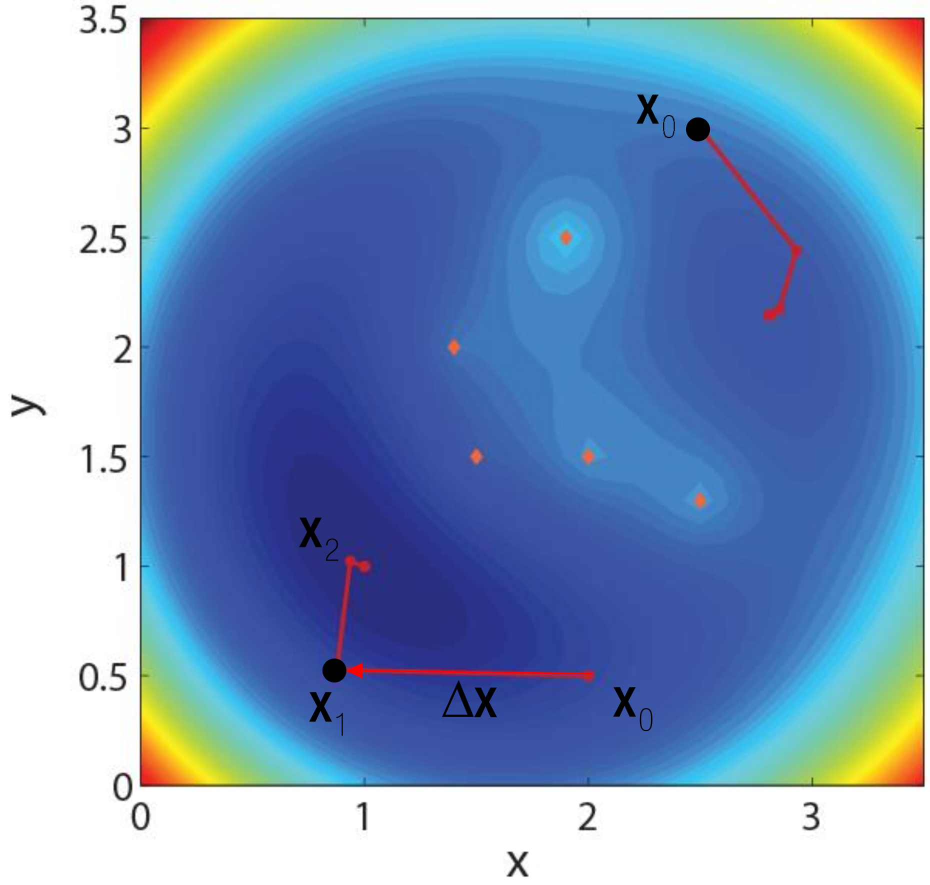
$$E = \left\| \underbrace{\begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix}}_{f(\mathbf{x})} - \underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix}}_{\mathbf{b}} \right\|^2$$

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Delta\mathbf{x} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} (\mathbf{b} - f(\mathbf{x}))$$

$$\Delta\mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} (\mathbf{b} - f(\mathbf{x}))$$

Where am I?: GPS Localization

Example:
Localization using range data from beacons



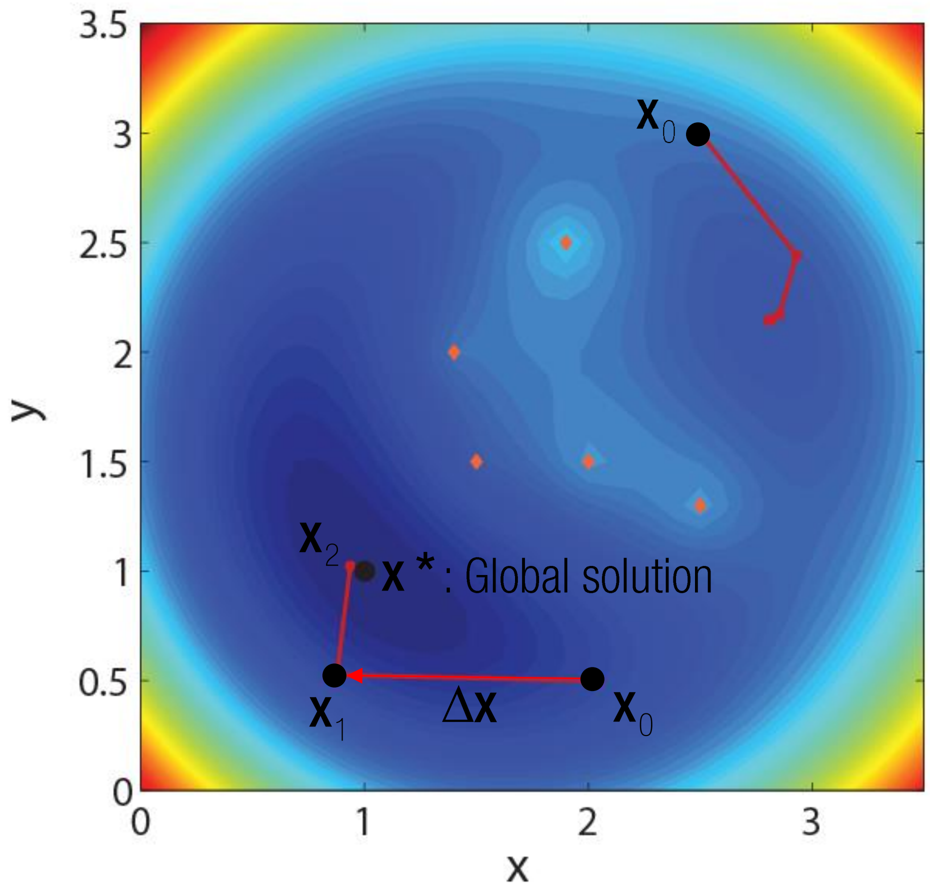
$$E = \left\| \underbrace{\begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix}}_{f(\mathbf{x})} - \underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix}}_{\mathbf{b}} \right\|^2$$

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Delta\mathbf{x} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

$$\Delta\mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

Where am I?: GPS Localization

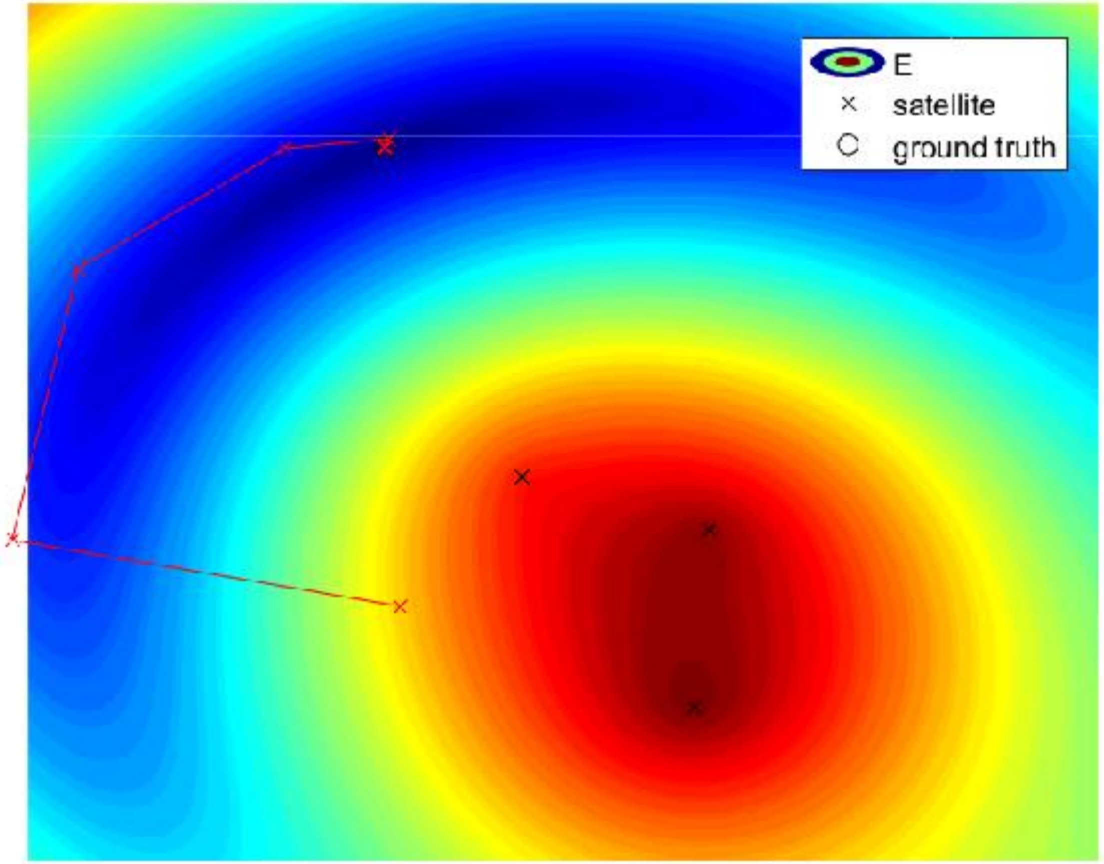
Example:
Localization using range data from beacons



$$E = \left\| \underbrace{\begin{bmatrix} \sqrt{(u_1 - x)^2 + (v_1 - y)^2} \\ \vdots \\ \sqrt{(u_5 - x)^2 + (v_5 - y)^2} \end{bmatrix}}_{f(\mathbf{x})} - \underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_5 \end{bmatrix}}_{\mathbf{b}} \right\|^2$$

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Delta\mathbf{x} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

$$\Delta\mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

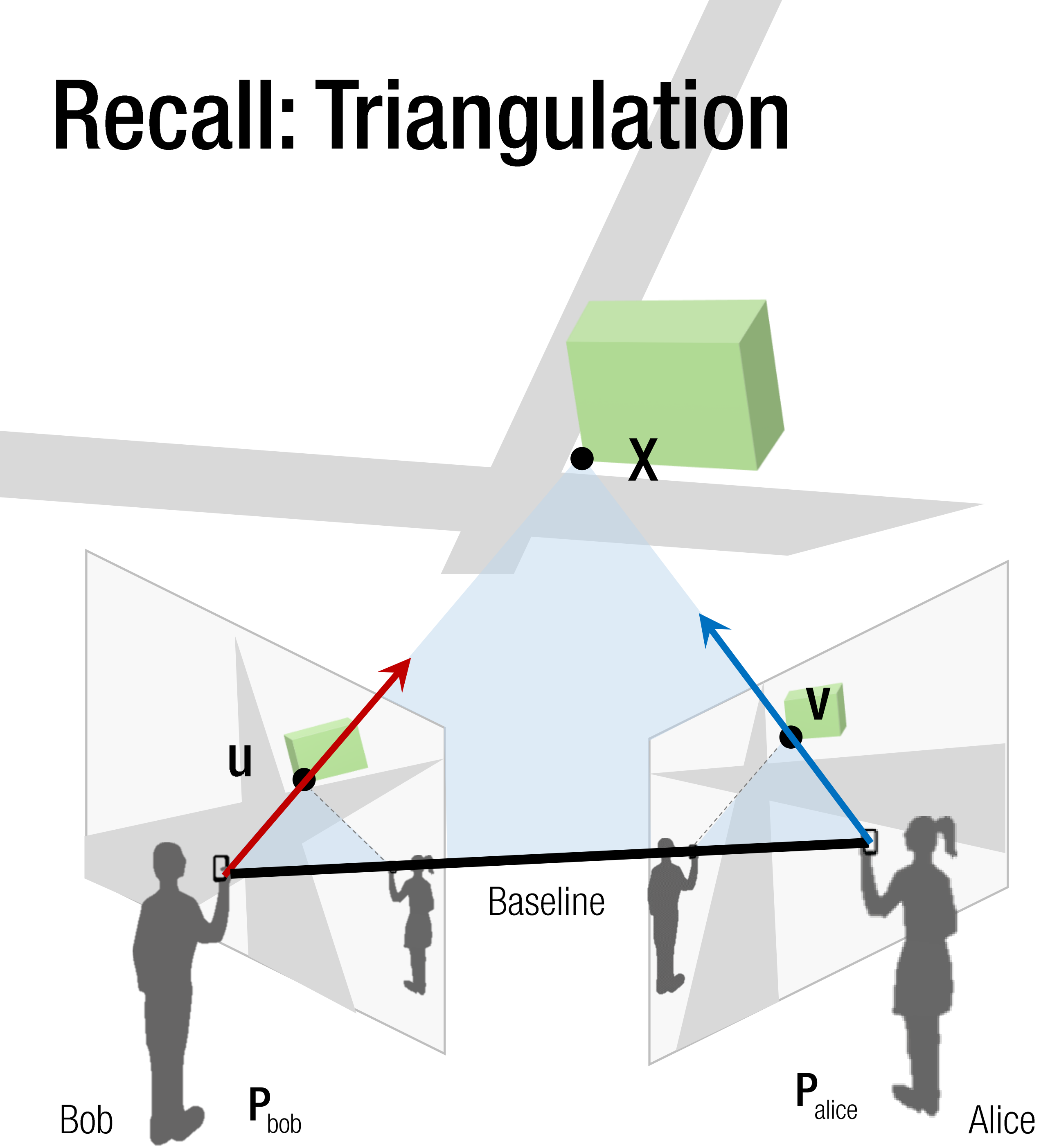


Beacon.m

```
u = 2*(rand(3,2)-0.5);
x = [-0.5 1];

for i = 1 : size(u,1)
   d(i,1) = norm(x-u(i,:));
end

[x_grid, y_grid] = meshgrid(-1.5:0.01:1.5, -1.5:0.01:1.5);

E = zeros(size(x_grid));
for i = 1 : size(u,1)
   E = E + (sqrt((x_grid-u(i,1)).^2 +(y_grid-u(i,2)).^2)-d(i)).^2 ;
end
E = sqrt(E);

x0 = 2*(rand(2,1)-0.5);
for j = 1 : 10
   J = [];
   fx = [];
   for i = 1 : size(u,1)
      denom = norm(u(i,:)-x0');
      J = [J; (u(i,:)-x0')/denom];
      fx = [fx; norm(u(i,:)-x0')];
   end
   b = d;
   delta_x = -inv(J'*J)*J'*(b-fx);
   x0 = x0 + delta_x;
end
```

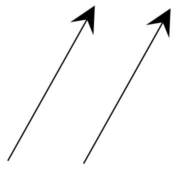
Triangulation Refinement

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} = \mathbf{P}_{bob} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

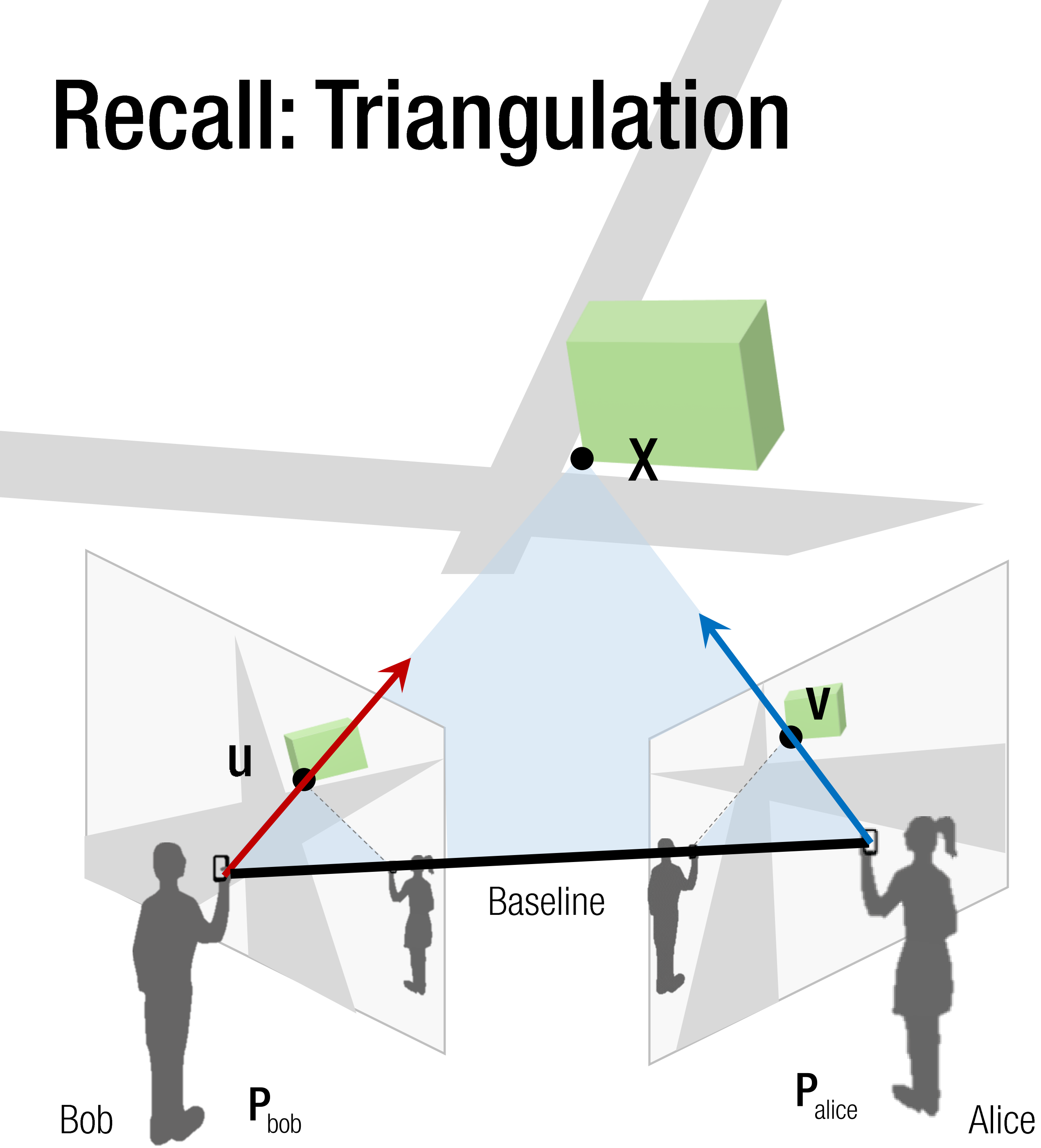


Two 3D vectors are parallel.

$$\longrightarrow \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} \times \mathbf{P}_{bob} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

$$\longrightarrow \underbrace{\begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix}_{\times}}_{\text{Skew-symmetric matrix}} \mathbf{P}_{bob} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} = \mathbf{P}_{bob} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

Two 3D vectors are parallel.

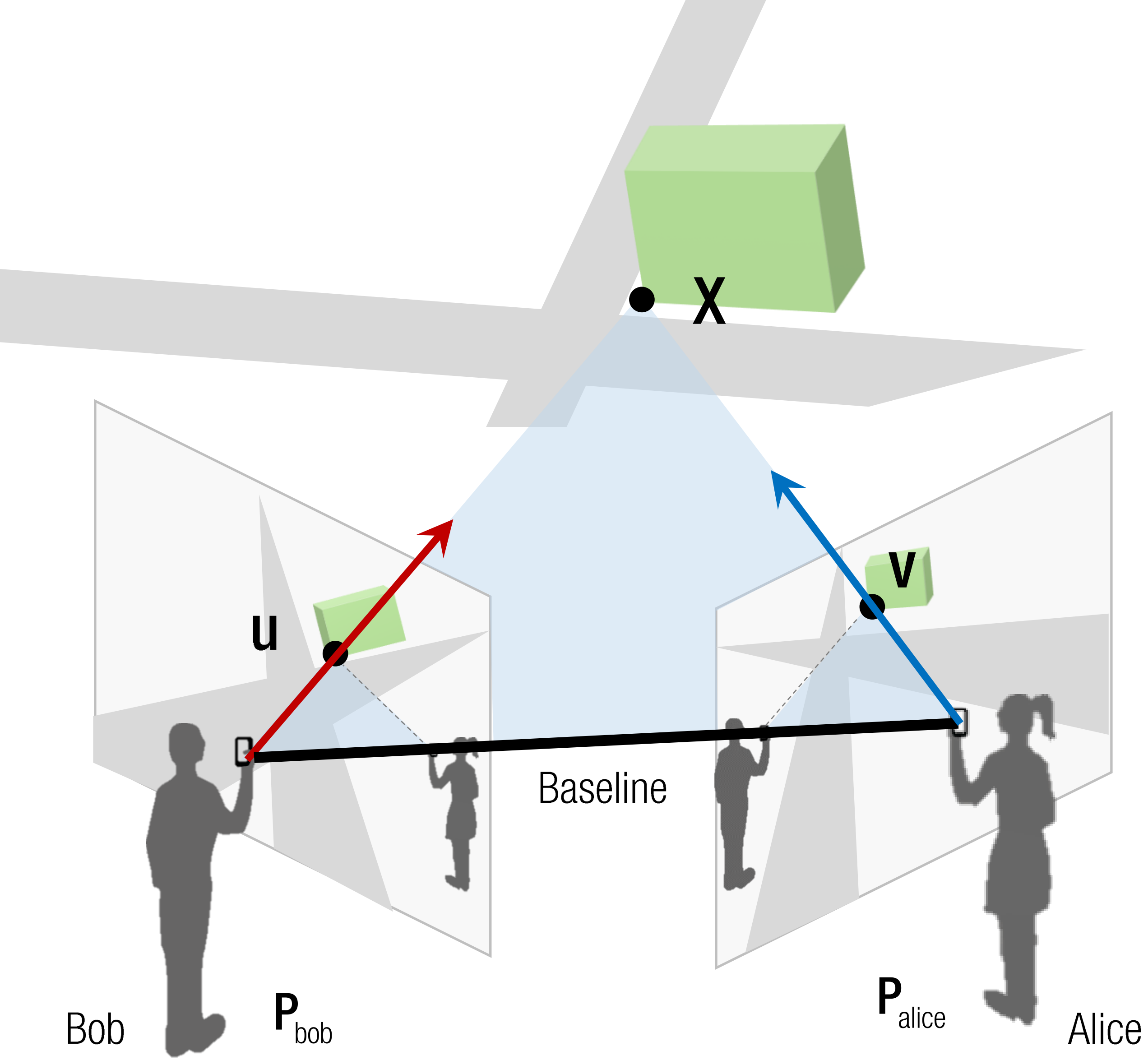
$$\rightarrow \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} \times \mathbf{P}_{bob} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

$$\rightarrow \underbrace{\begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix}_{\times}}_{\text{Skew-symmetric matrix}} \mathbf{P}_{bob} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

: Knowns

: Unknowns

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} = \mathbf{P}_{\text{bob}} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

Two 3D vectors are parallel.

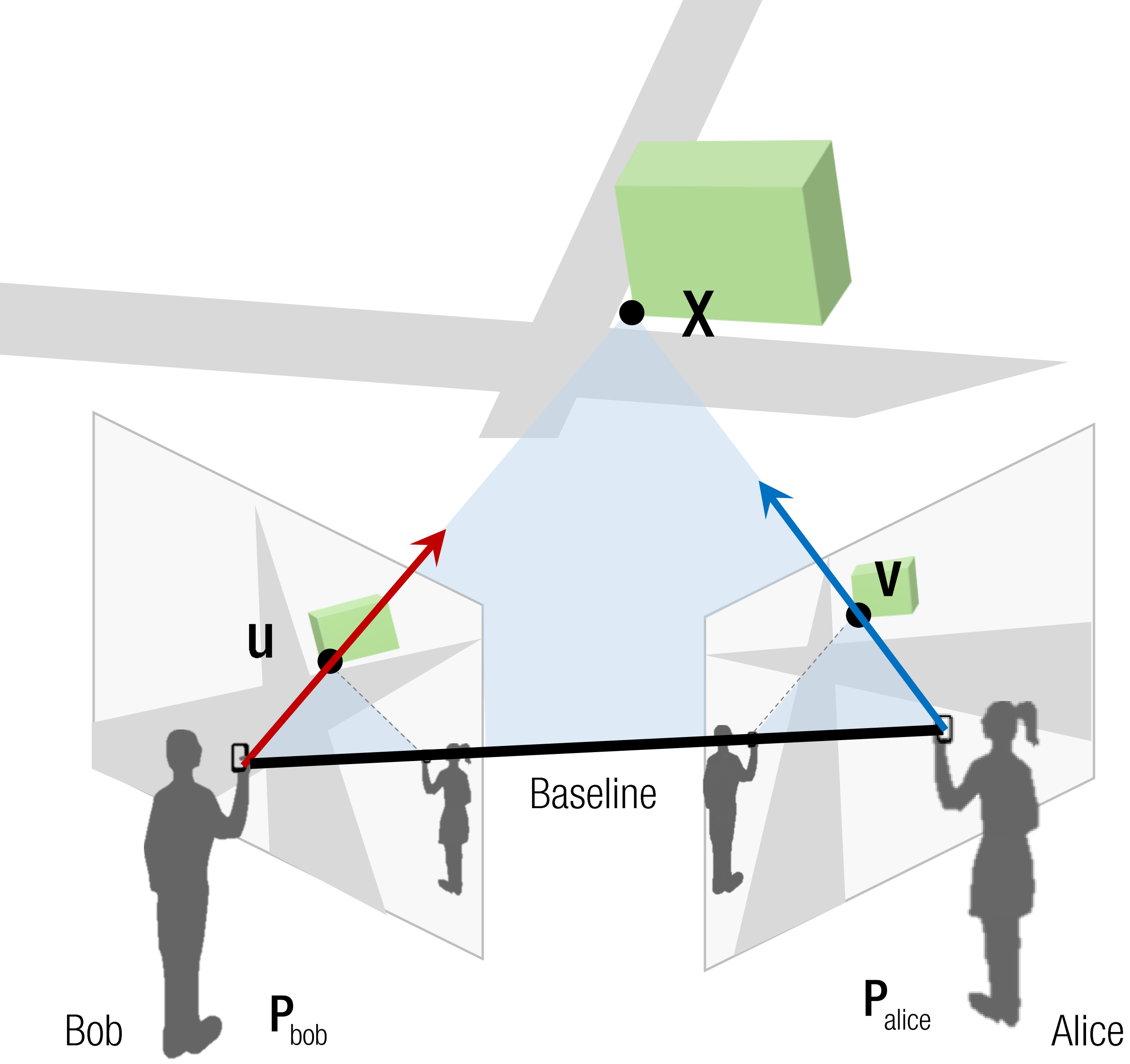
$$\rightarrow \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} \times \mathbf{P}_{\text{bob}} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

$$\rightarrow \underbrace{\begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix}_{\times} \mathbf{P}_{\text{bob}}}_{2\text{x}4} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

: Knowns

: Unknowns

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} = \mathbf{P}_{bob} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

Two 3D vectors are parallel.

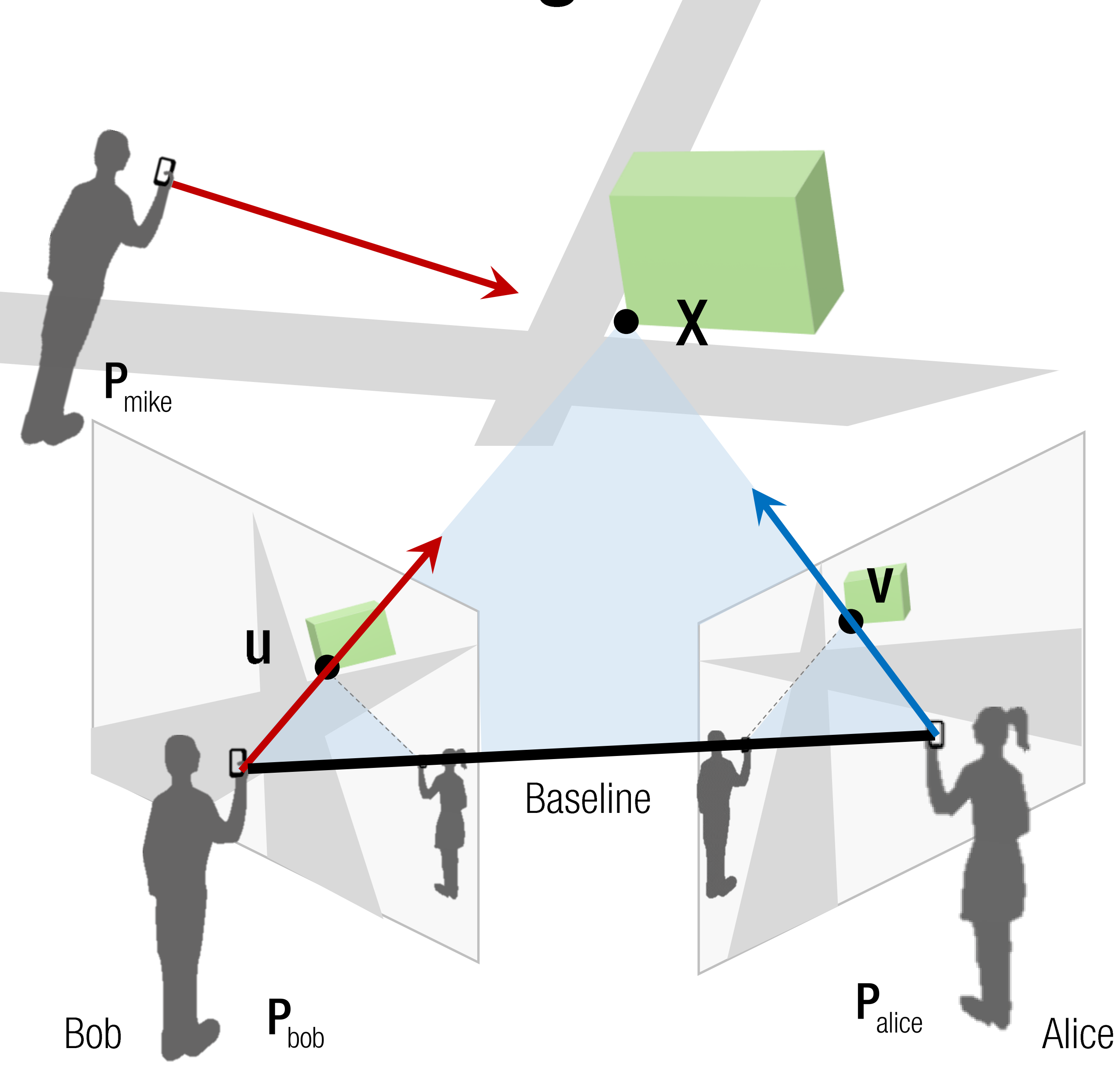
$$\rightarrow \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} \times \mathbf{P}_{bob} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

$$\rightarrow \underbrace{\begin{bmatrix} \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix}_\times \mathbf{P}_{bob} \\ \begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}_\times \mathbf{P}_{alice} \end{bmatrix}}_{4\text{x}4} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

: Knowns

: Unknowns

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} = \mathbf{P}_{\text{bob}} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

Two 3D vectors are parallel.

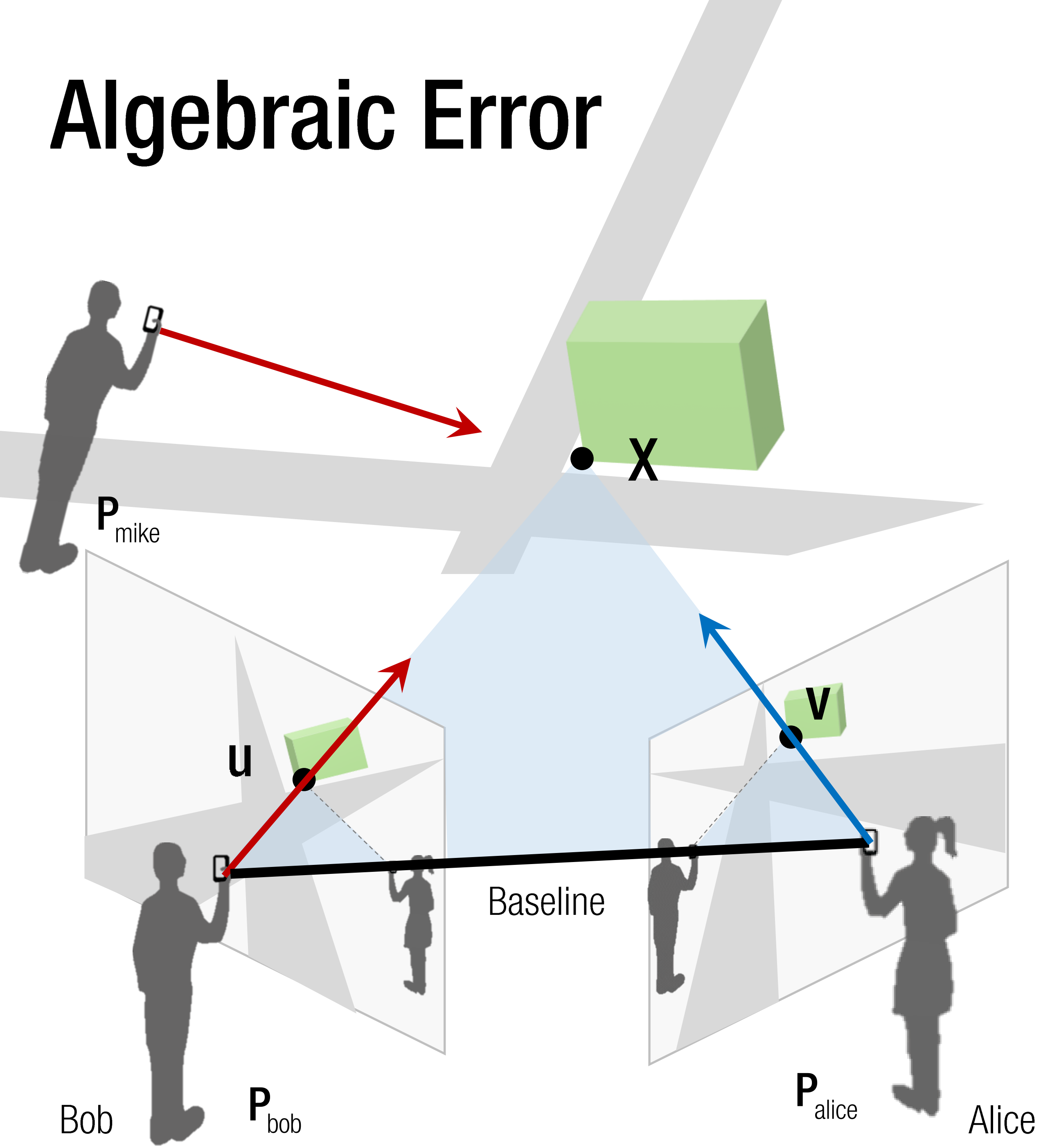
$$\rightarrow \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} \times \mathbf{P}_{\text{bob}} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

$$\rightarrow \begin{matrix} \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix}_\times \mathbf{P}_{\text{bob}} \\ \begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}_\times \mathbf{P}_{\text{alice}} \\ \begin{bmatrix} \mathbf{w} \\ 1 \end{bmatrix}_\times \mathbf{P}_{\text{mike}} \end{matrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \mathbf{0}$$

: Knowns

: Unknowns

Algebraic Error



$$E_{alge} = \left\| \mathbf{A}\,\mathbf{x} - \mathbf{b} \right\|^2$$

Algebraic error does not have geometric meaning.

Recall: Geometric Verification via Reprojection Error

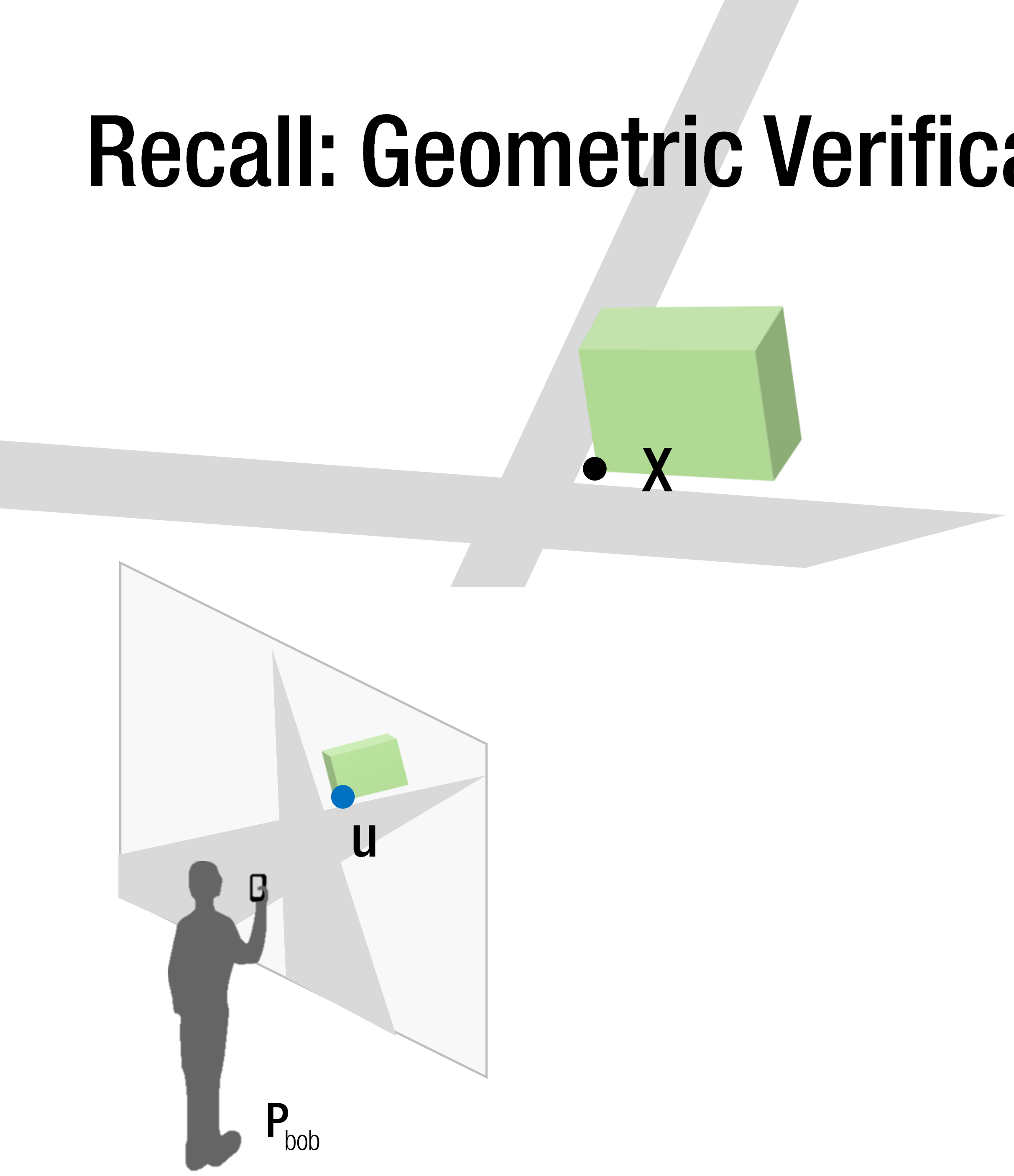


Image feature measurement, e.g. SIFT detection:

u

Recall: Geometric Verification via Reprojection Error

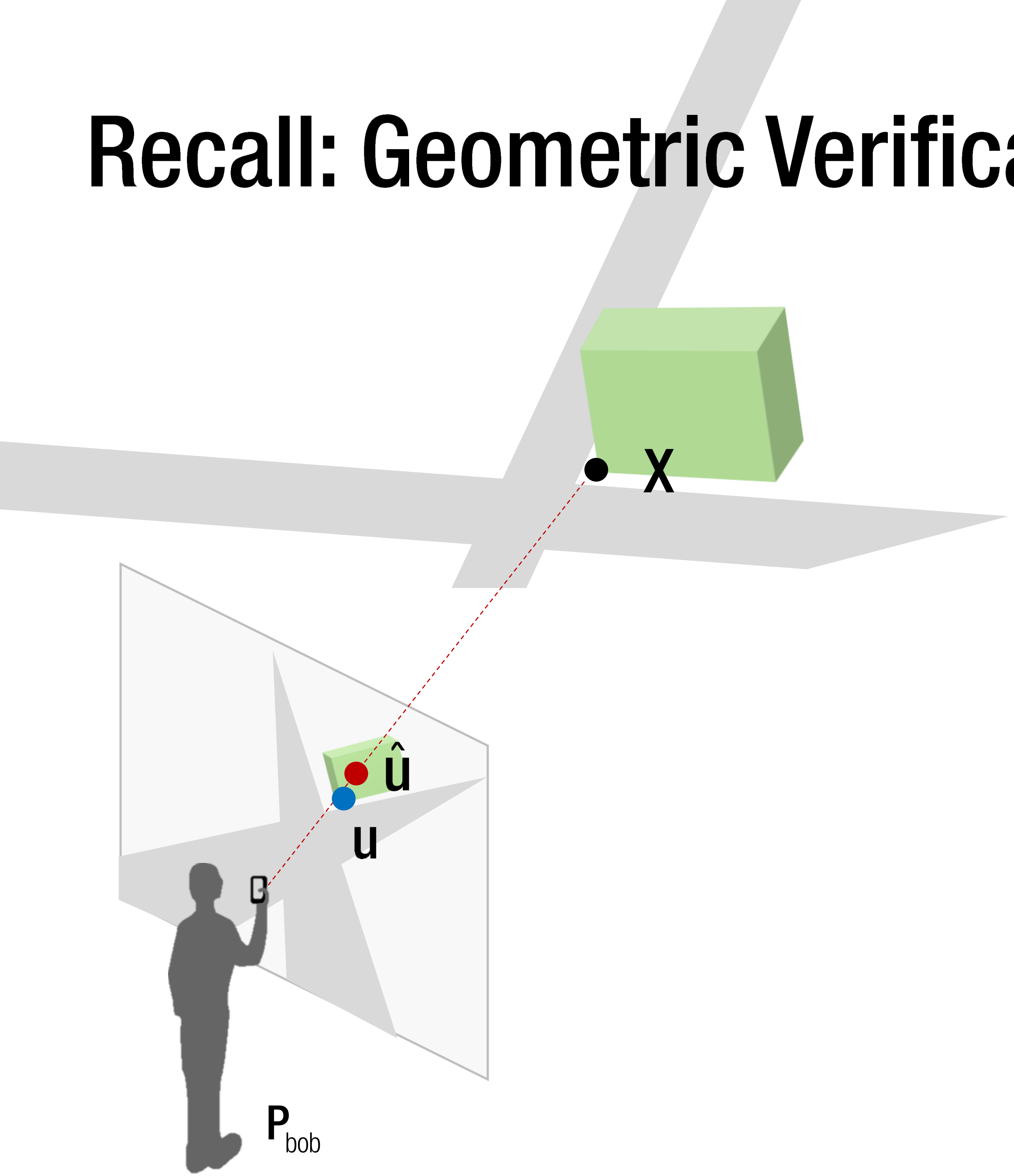


Image feature measurement, e.g. SIFT detection:

$\mathbf{u}$

3D point projection, or reprojection:

$$\lambda \hat{\mathbf{u}} = \mathbf{P}\mathbf{X}$$

Recall: Geometric Verification via Reprojection Error

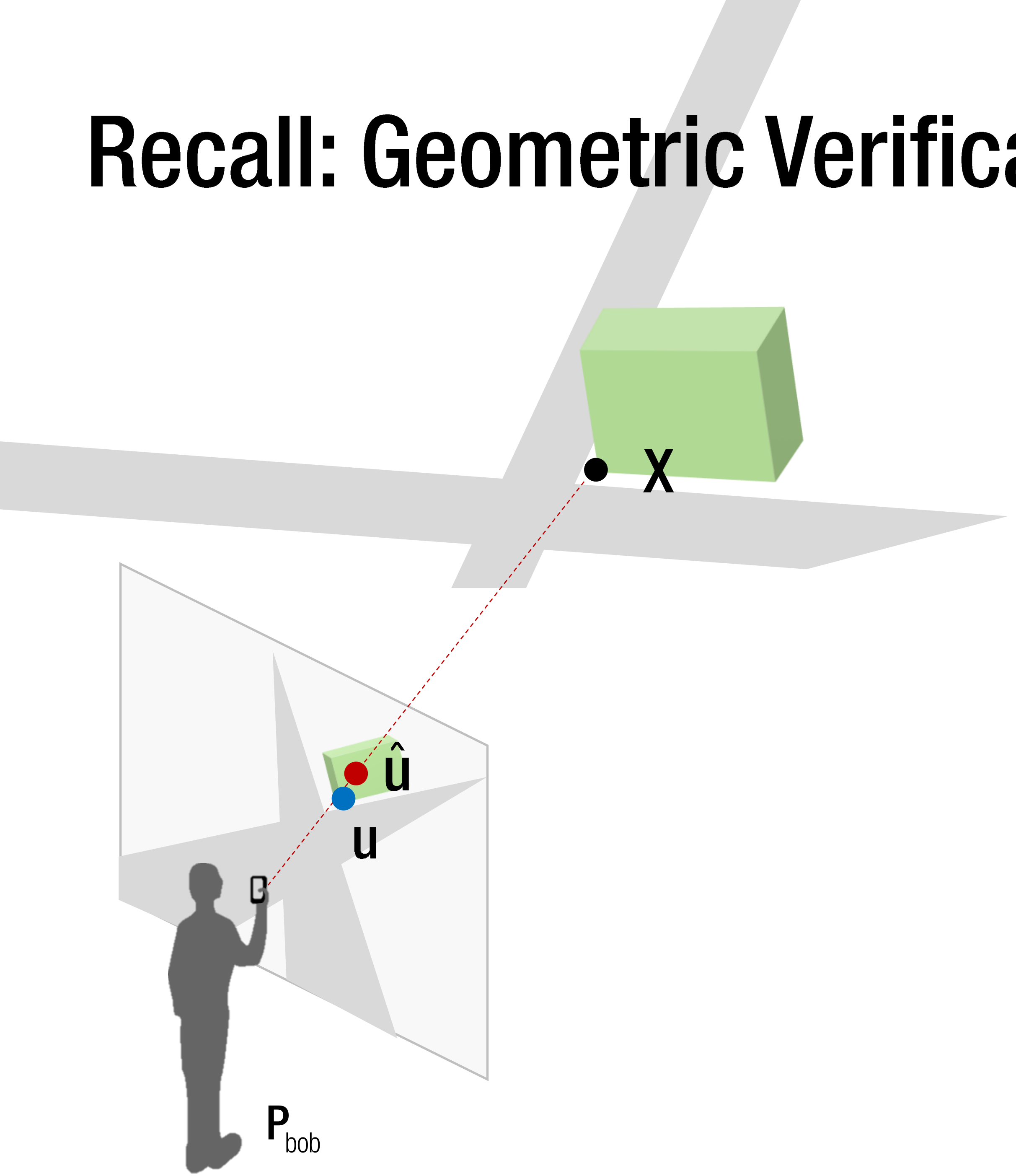


Image feature measurement, e.g. SIFT detection:

$\mathbf{u}$

3D point projection, or reprojection:

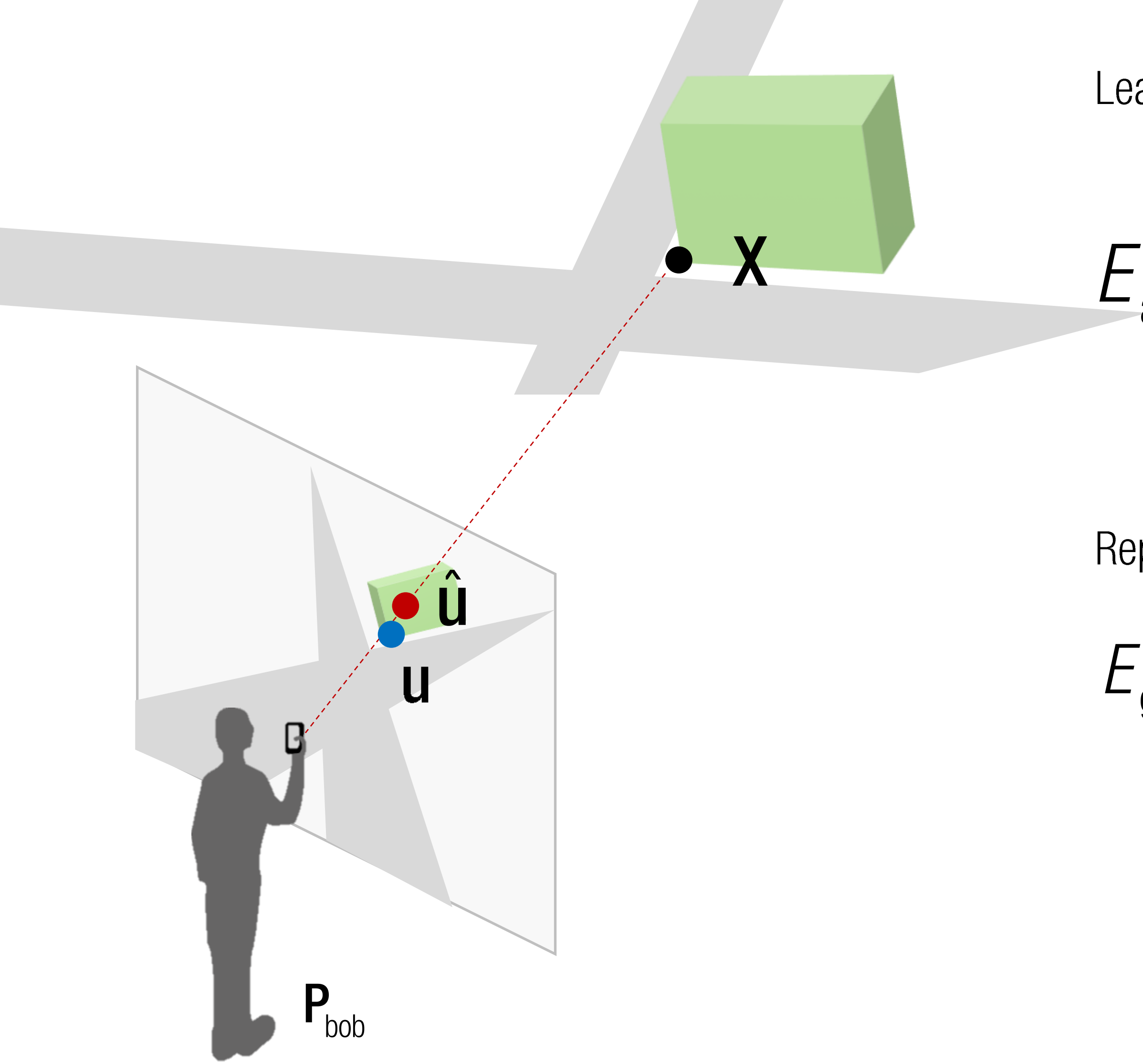
$$\lambda \hat{\mathbf{u}} = \mathbf{P}\mathbf{X}$$

Reprojection error (geometric error):

$$E_{geom} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

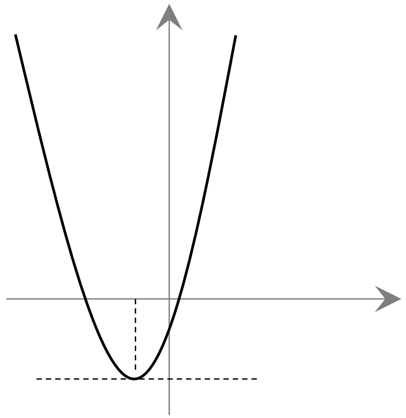
$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - \mathbf{u}_1 \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - \mathbf{u}_2 \right)^2$$

Algebraic vs. Geometric error



Least squares solution (algebraic error):

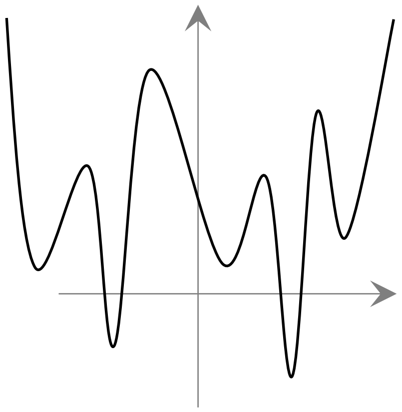
$$E_{alge} = \left\| \mathbf{A}\,\mathbf{x} - \mathbf{b} \right\|^2$$



Reprojection error (geometric error):

$$E_{geom} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - \mathbf{u}_1 \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - \mathbf{u}_2 \right)^2$$

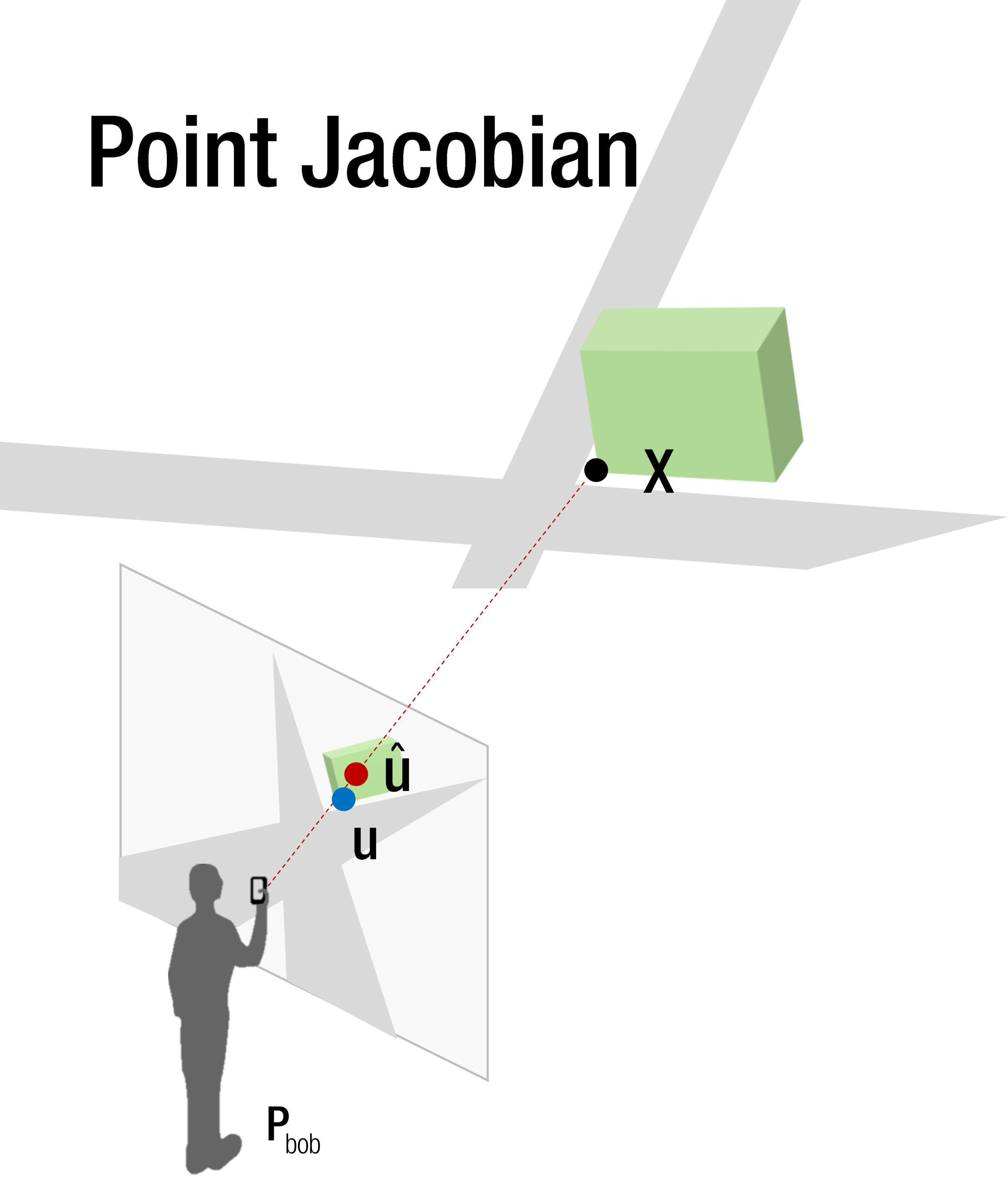


Point Jacobian

Black: given variables
Red: unknowns

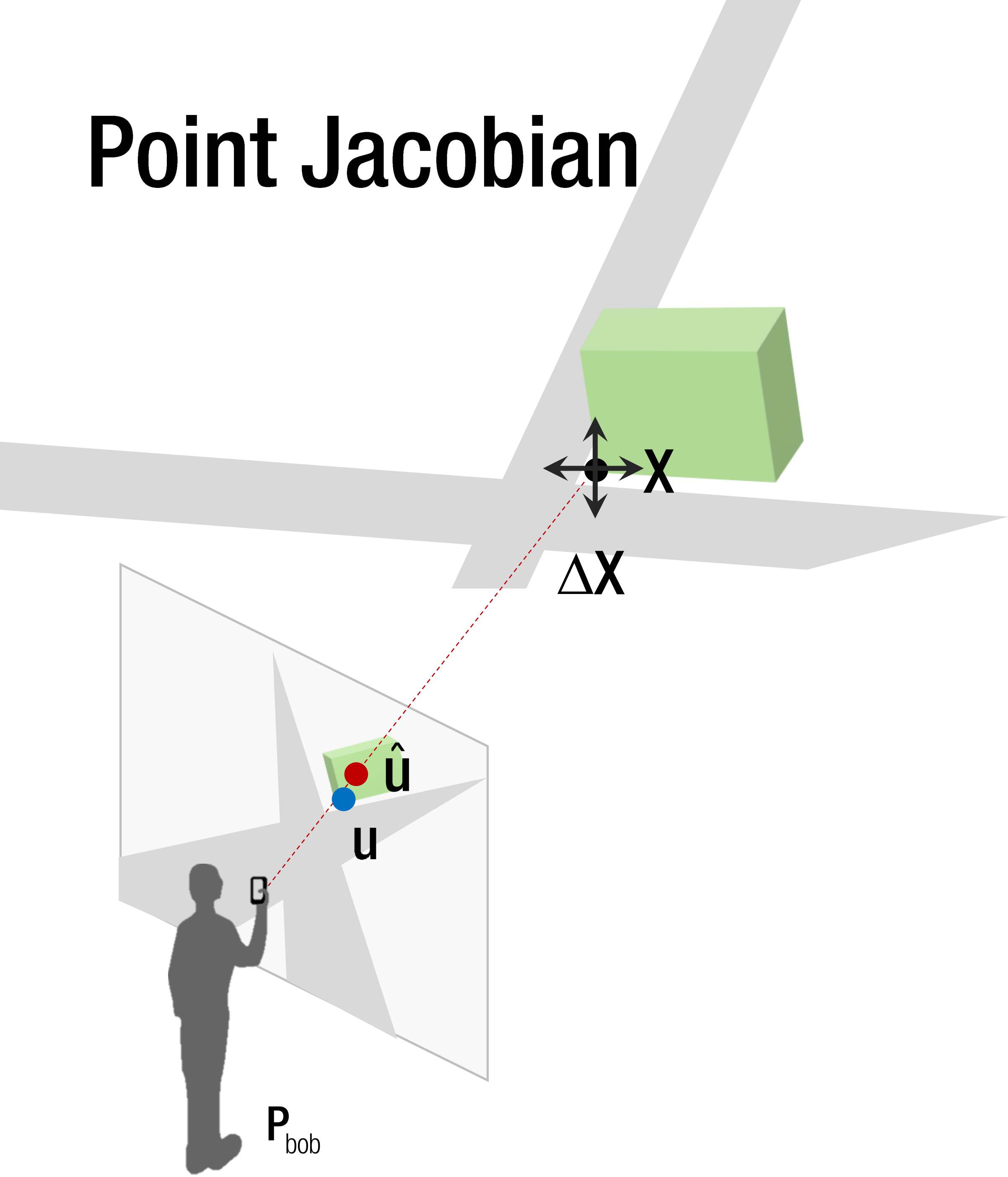
$$E_{\text{geom}} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - y \right)^2$$



Black: given variables
Red: unknowns

Point Jacobian



$$E_{geom} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

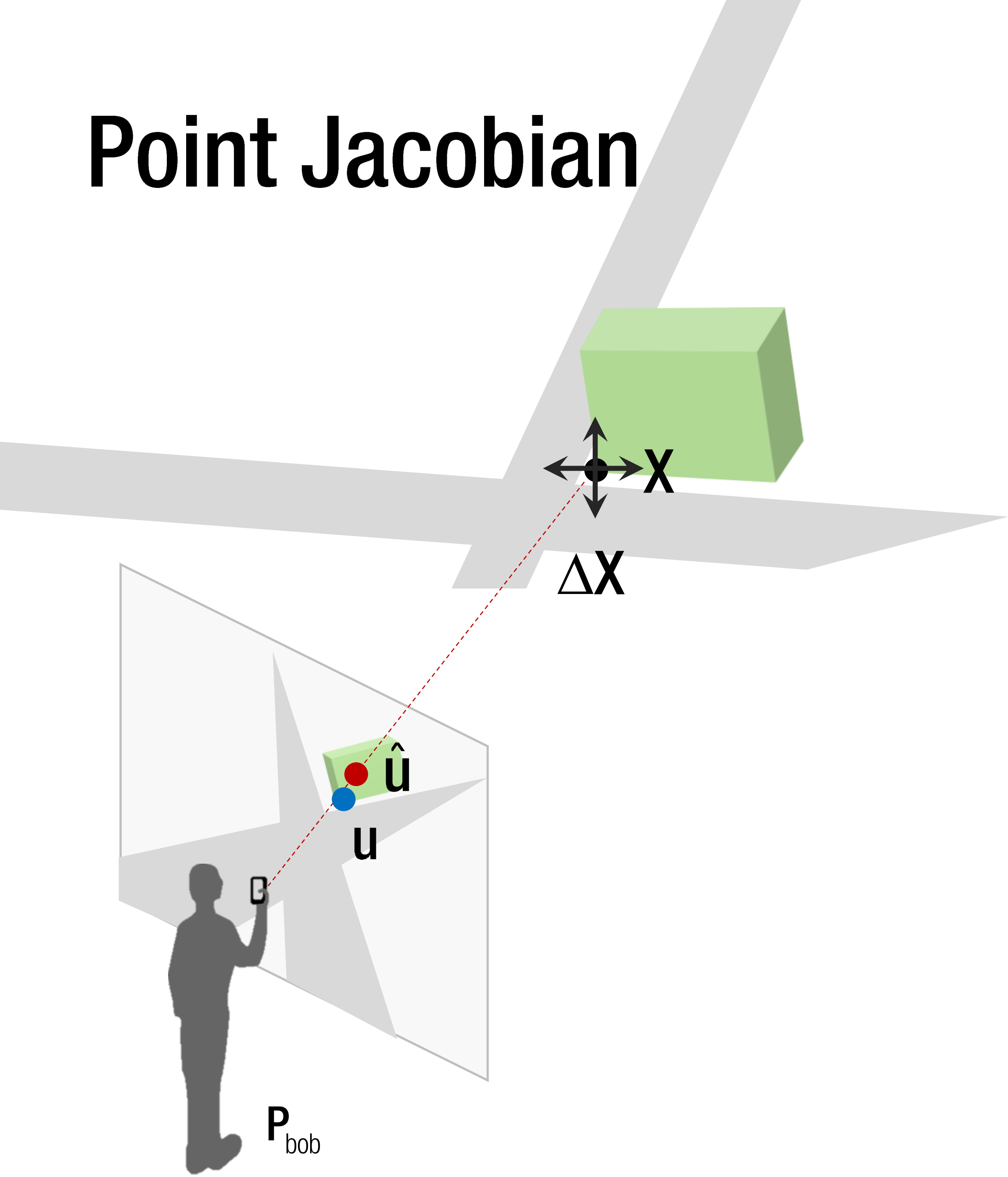
$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - y \right)^2$$

$$\Delta\mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x}) \right)$$

Point Jacobian

Black: given variables
Red: unknowns

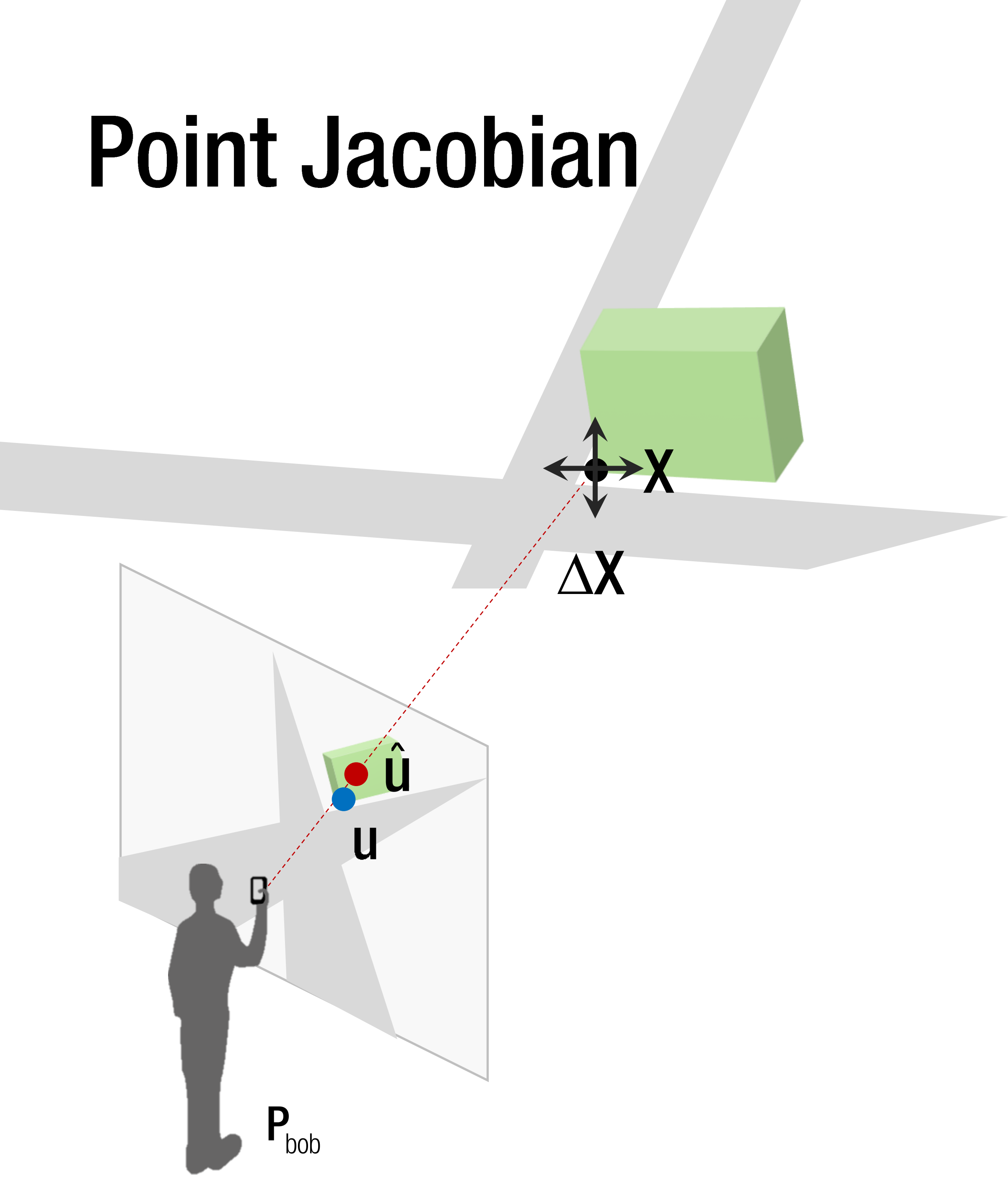
$$E_{geom} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X} - \mathbf{C})$$



$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

Black: given variables
Red: unknowns

Point Jacobian

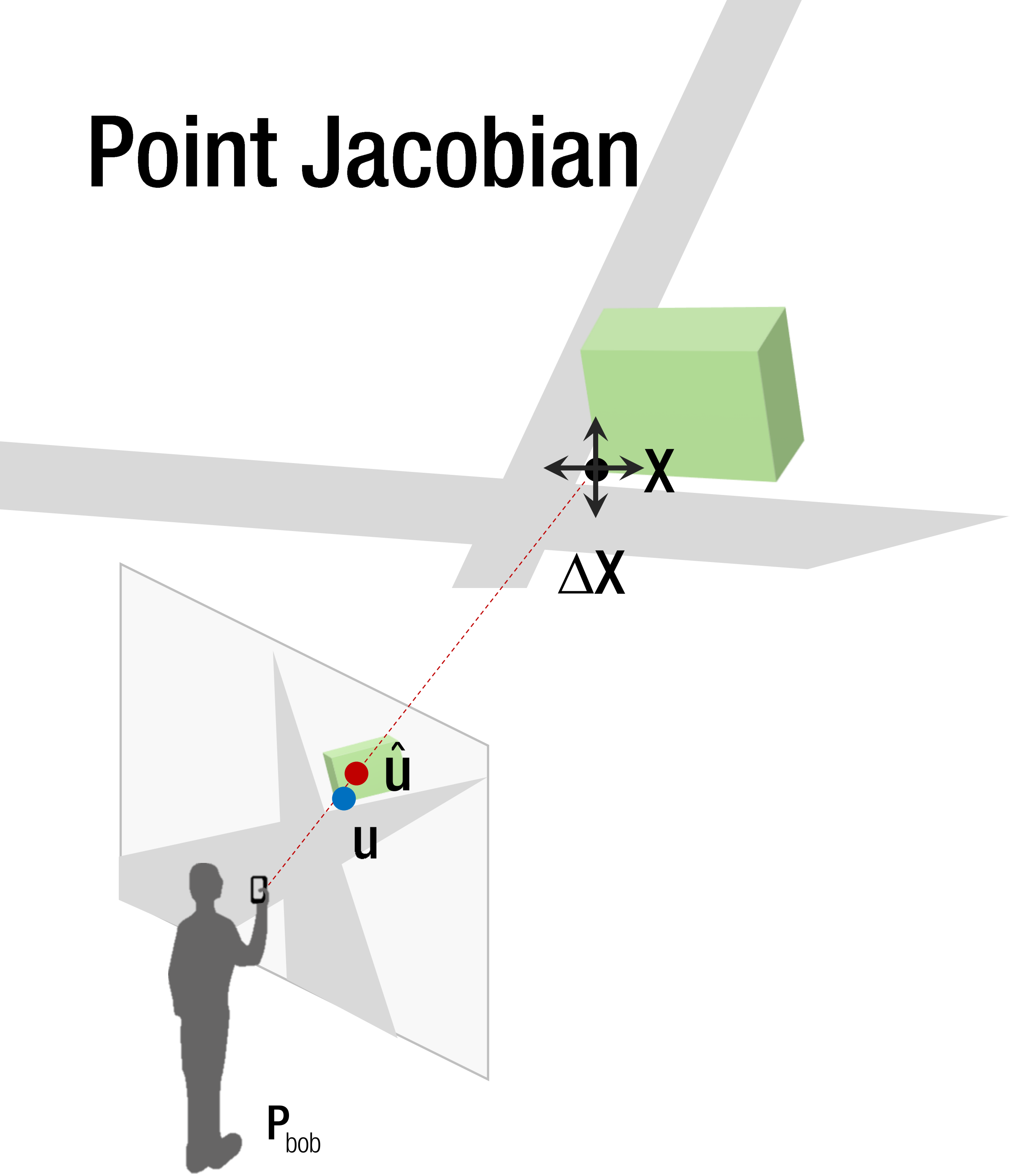


$$E_{geom} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X} - \mathbf{C})$$

$$f(\mathbf{X}) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix}$$

$$\Delta\mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathrm{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

Point Jacobian



Black: given variables
Red: unknowns

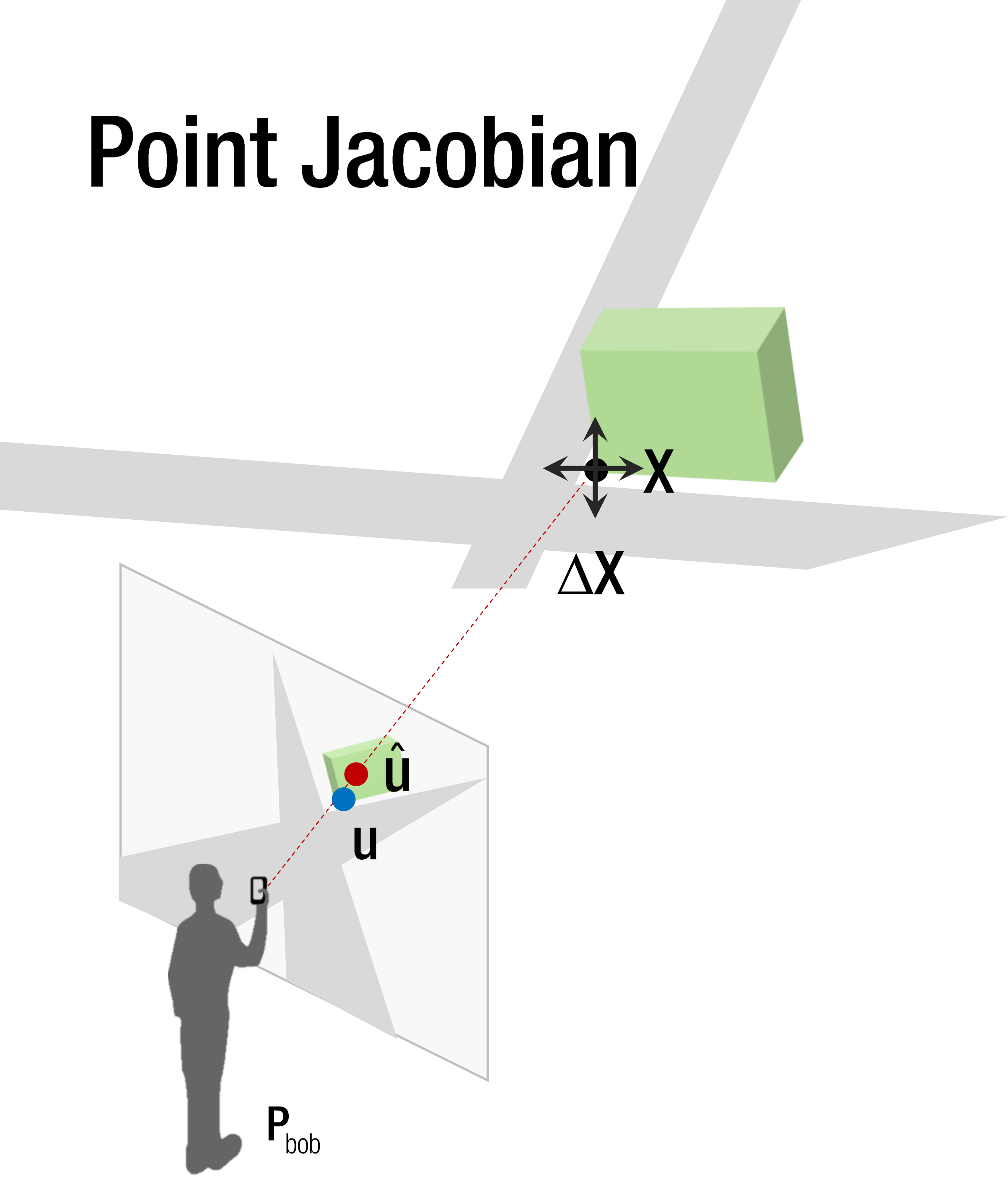
$$E_{\text{geom}} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X} - \mathbf{C})$$

$$f(\mathbf{X}) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} \longrightarrow \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w \frac{\partial u}{\partial \mathbf{X}} - u \frac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \dfrac{v \frac{\partial u}{\partial \mathbf{X}} - v \frac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix}$$

$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

Point Jacobian

Black: given variables
Red: unknowns



$$E_{geom} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X} - \mathbf{C})$$

$$\rightarrow \quad \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}$$

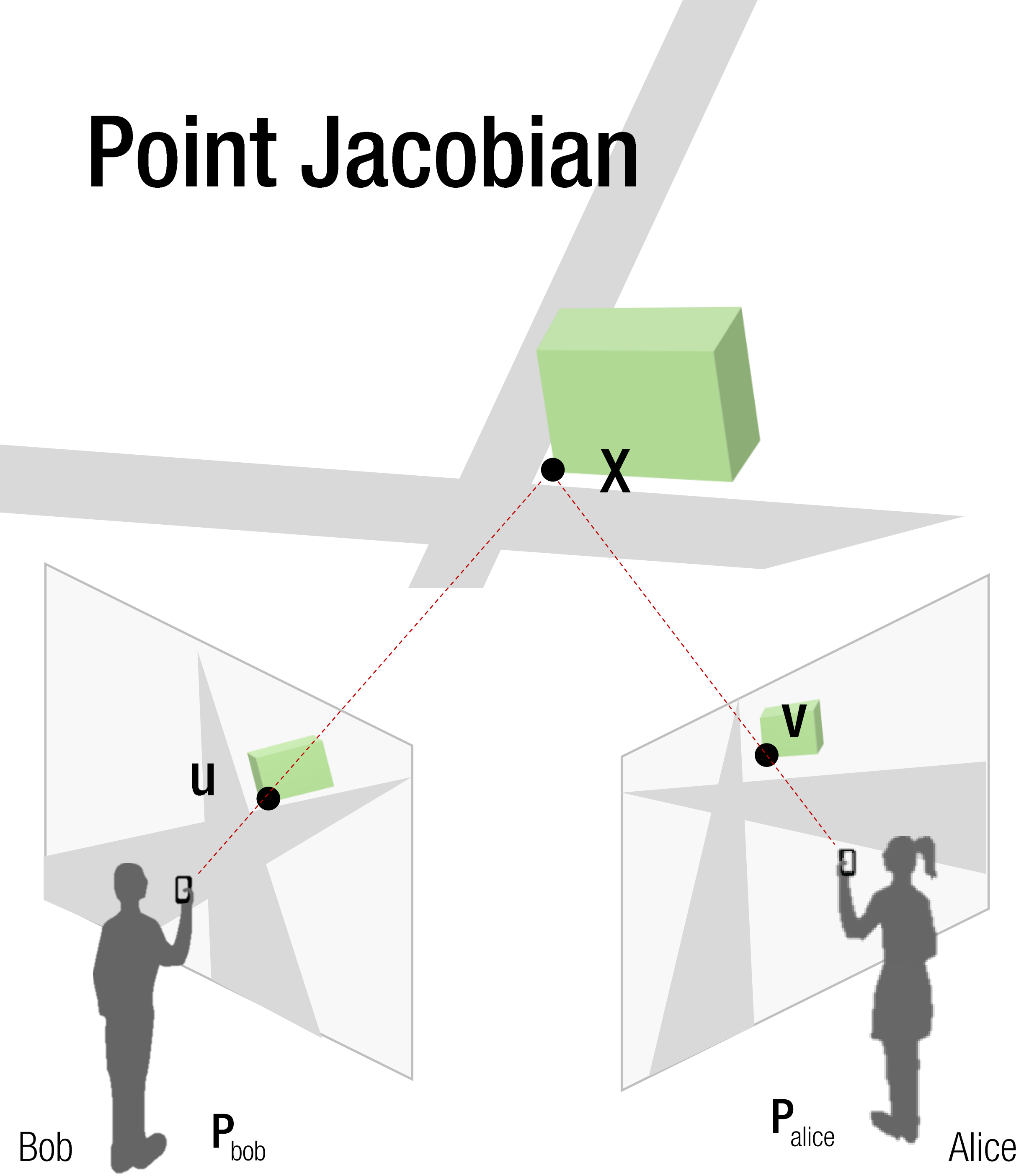
$$f(\mathbf{X}) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} \quad \rightarrow \quad \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\frac{\partial u}{\partial \mathbf{X}} - u\frac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \dfrac{v\frac{\partial u}{\partial \mathbf{X}} - v\frac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix}$$

$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x})\right)$$

Point Jacobian

Black: given variables

Red: unknowns



$$E_{\text{geom}} = \left\| \begin{bmatrix} u_{\text{bob}} / w_{\text{bob}} \\ v_{\text{bob}} / w_{\text{bob}} \\ u_{\text{alice}} / w_{\text{alice}} \\ v_{\text{alice}} / w_{\text{alice}} \end{bmatrix} - \begin{bmatrix} x_{\text{bob}} \\ y_{\text{bob}} \\ x_{\text{alice}} \\ y_{\text{alice}} \end{bmatrix} \right\|^2$$

$$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\frac{\partial u}{\partial \mathbf{X}} - u\frac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \dfrac{v\frac{\partial u}{\partial \mathbf{X}} - v\frac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix}$$

$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x}) \right)$$

Algorithm 3 Nonlinear Point Refinement

b $= \begin{bmatrix} \mathbf{u}_1^\mathsf{T} & \mathbf{u}_2^\mathsf{T} \end{bmatrix}^\mathsf{T}$
for j = 1 : nIters **do**
    Build point Jacobian, $\frac{\partial f(\mathbf{X})_j}{\partial \mathbf{X}}$.
    Compute $f(\mathbf{X})$.
    $\Delta\mathbf{X} = \left(\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}^\mathsf{T} \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} + \lambda\mathbf{I}\right)^{-1} \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}^\mathsf{T} (\mathbf{b} - f(\mathbf{X}))$
    $\mathbf{X} = \mathbf{X} + \Delta\mathbf{X}$
end for

$$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \frac{w\frac{\partial u}{\partial \mathbf{X}} - u\frac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \frac{v\frac{\partial u}{\partial \mathbf{X}} - v\frac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix}$$

$$\Delta\mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\mathsf{T} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\mathsf{T} (\mathbf{b} - f(\mathbf{x}))$$

Algorithm 3 Nonlinear Point Refinement

$\mathbf{b} = \begin{bmatrix} \mathbf{u}_1^\mathsf{T} & \mathbf{u}_2^\mathsf{T} \end{bmatrix}^\mathsf{T}$

for j = 1 : nIters **do**

  Build point Jacobian, $\frac{\partial f(\mathbf{X})_j}{\partial \mathbf{X}}$.

  Compute $f(\mathbf{X})$.

  $\Delta \mathbf{X} = \left(\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}^\mathsf{T} \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} + \boxed{\lambda \mathbf{I}} \right)^{-1} \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}^\mathsf{T} (\mathbf{b} - f(\mathbf{X}))$

  $\mathbf{X} = \mathbf{X} + \Delta \mathbf{X}$

end for

$$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w \frac{\partial u}{\partial \mathbf{X}} - u \frac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \dfrac{v \frac{\partial u}{\partial \mathbf{X}} - v \frac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix}$$

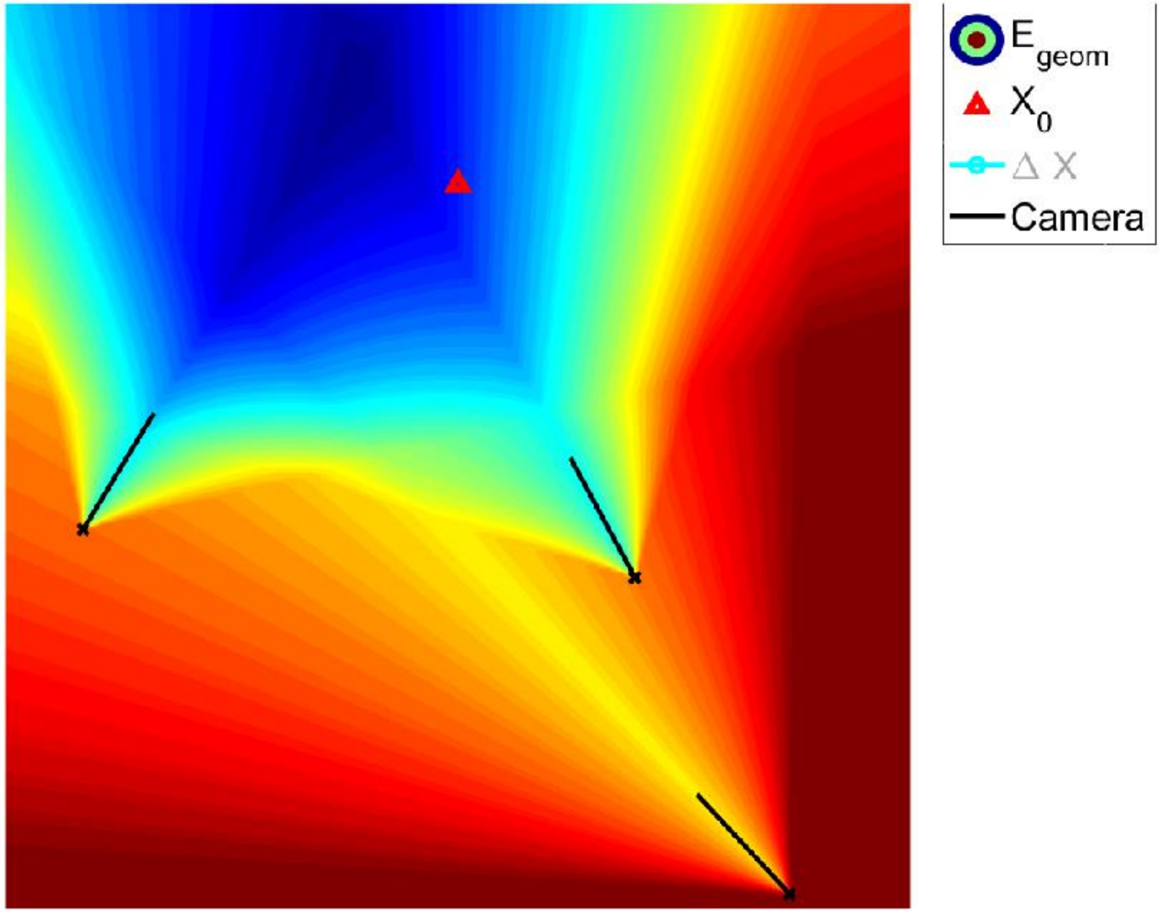
$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\mathsf{T} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\mathsf{T} (\mathbf{b} - f(\mathbf{x}))$$

Damping factor (Levenberg-Marquardt algorithm)

$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\mathsf{T} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} + \lambda \mathbf{I} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\mathsf{T} (\mathbf{b} - f(\mathbf{x}))$$

Black: given variables
Red: unknowns

Example: 1D Camera Triangulation

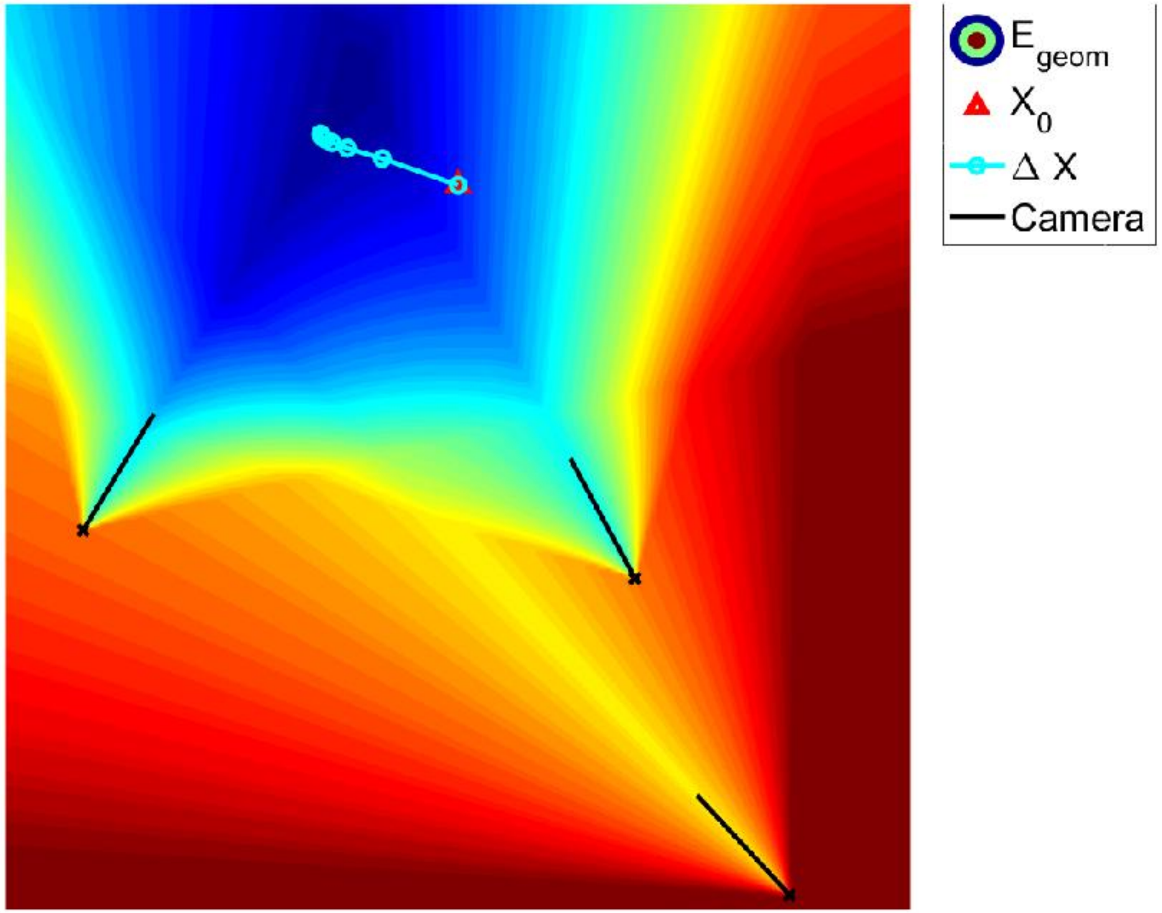



```
function df_dx = JacobianX_1D(K, R,  C,  X)

x = K * R * (X-C);

u = x(1);
w = x(2);

del = K * R;
du_dc = del(1,:);
dw_dc = del(2,:);

df_dx = [(w*du_dc-u*dw_dc)/w^2];
```

$$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \begin{bmatrix} \dfrac{w\dfrac{\partial u}{\partial \mathbf{X}} - u\dfrac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \dfrac{v\dfrac{\partial u}{\partial \mathbf{X}} - v\dfrac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix}$$

Black: given variables
Red: unknowns

Example: 1D Camera Triangulation

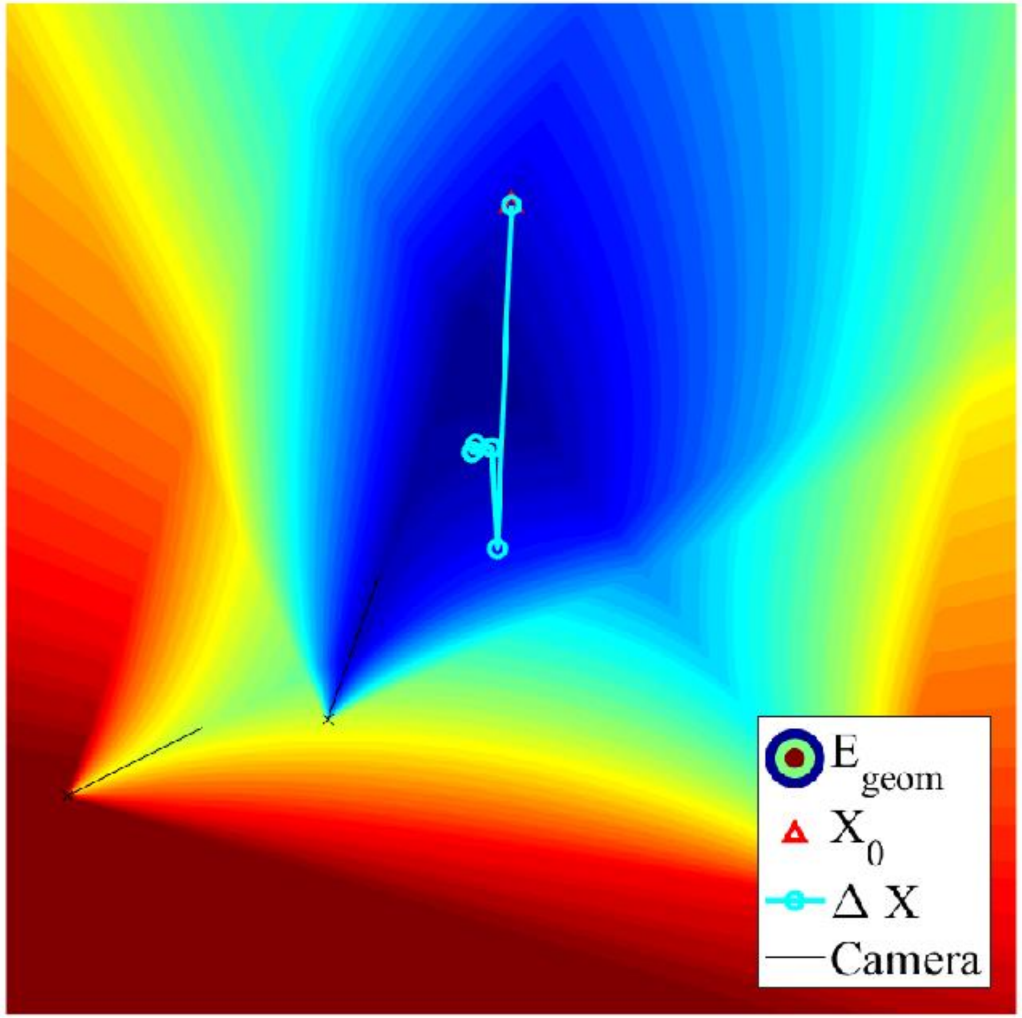


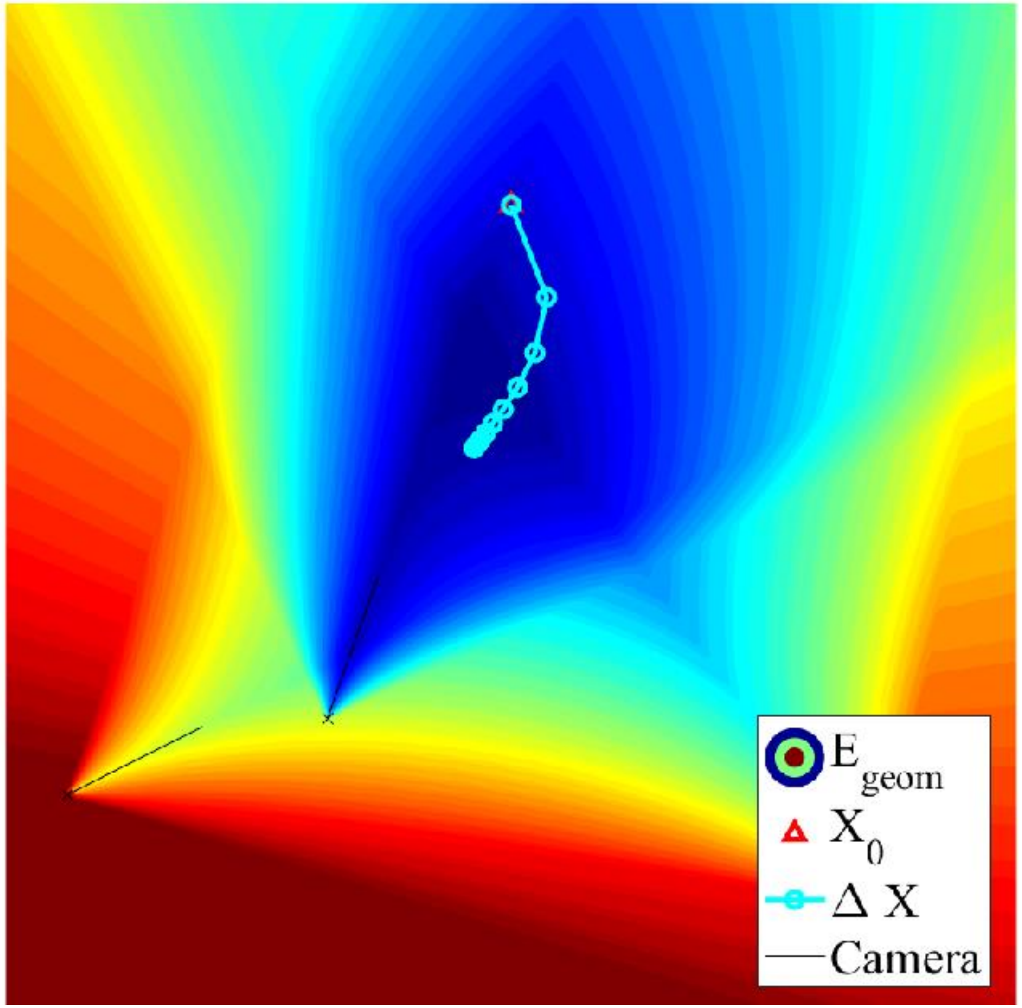
NonlinearTriangulation1D.m

```
for j = 1 : 10
    df_dX = [];
    delta_b = [];
    for i = 1 : size(c,2)
        df_dX = [df_dX; JacobianX(eye(2), R{i}, c(:,i), x)];
        u = R{i} * (x-c(:,i));
        delta_b = [delta_b; -u(1)/u(2)];
    end

    jacobian = df_dX;

    norm(delta_b)

    delta_x =
inv(jacobian'*jacobian+lambda*eye(size(jacobian'*jacobian,1)))*jacobia
n'*delta_b;
    x = x + delta_x;
    X(:,j+1) = x;
end
```

$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} + \lambda \mathbf{I} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x}) \right)$$

Damping Factor



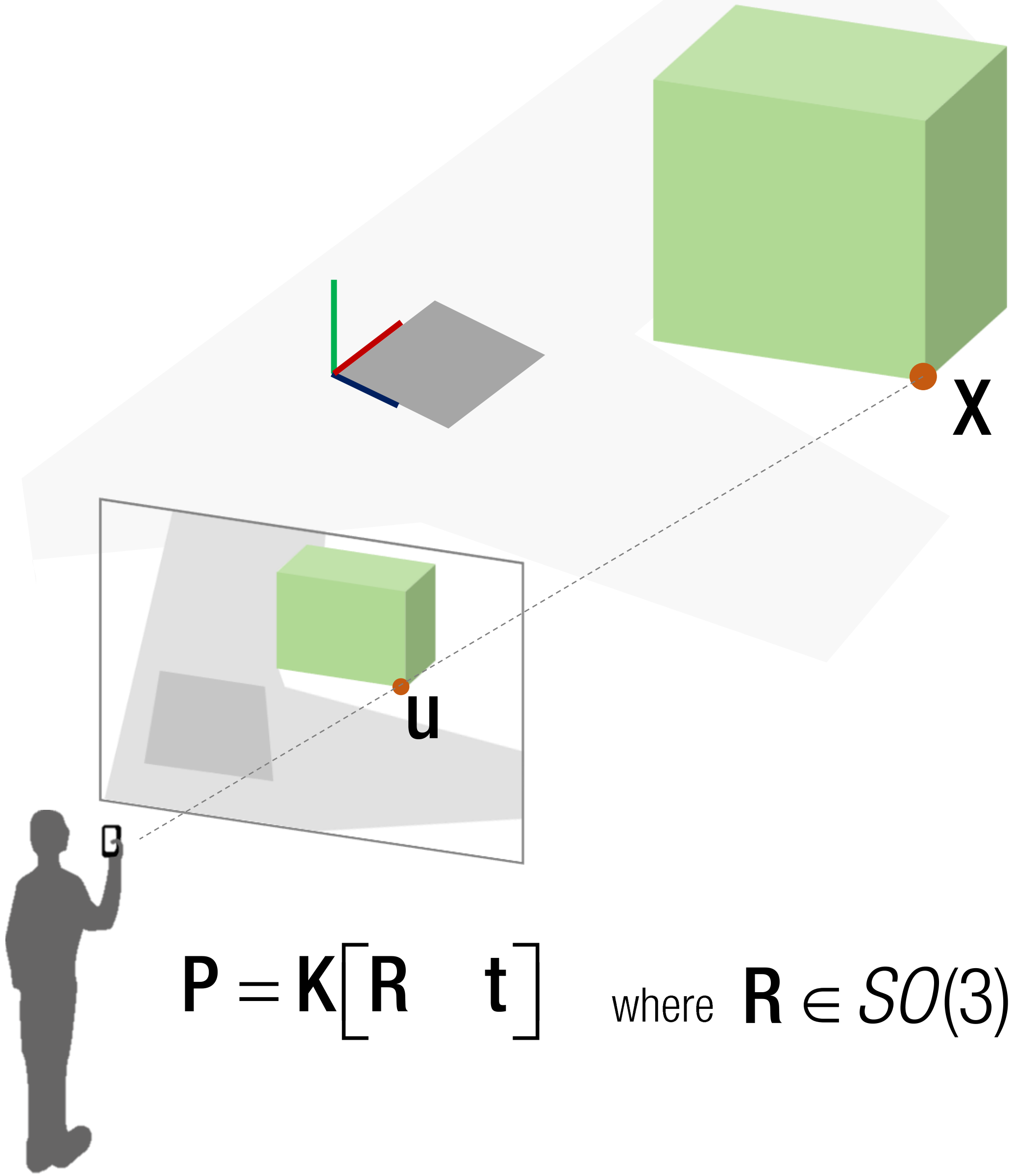


$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x}) \right)$$

$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} + \lambda \mathbf{I} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{x}) \right)$$

PnP Refinement

3D-2D Correspondence



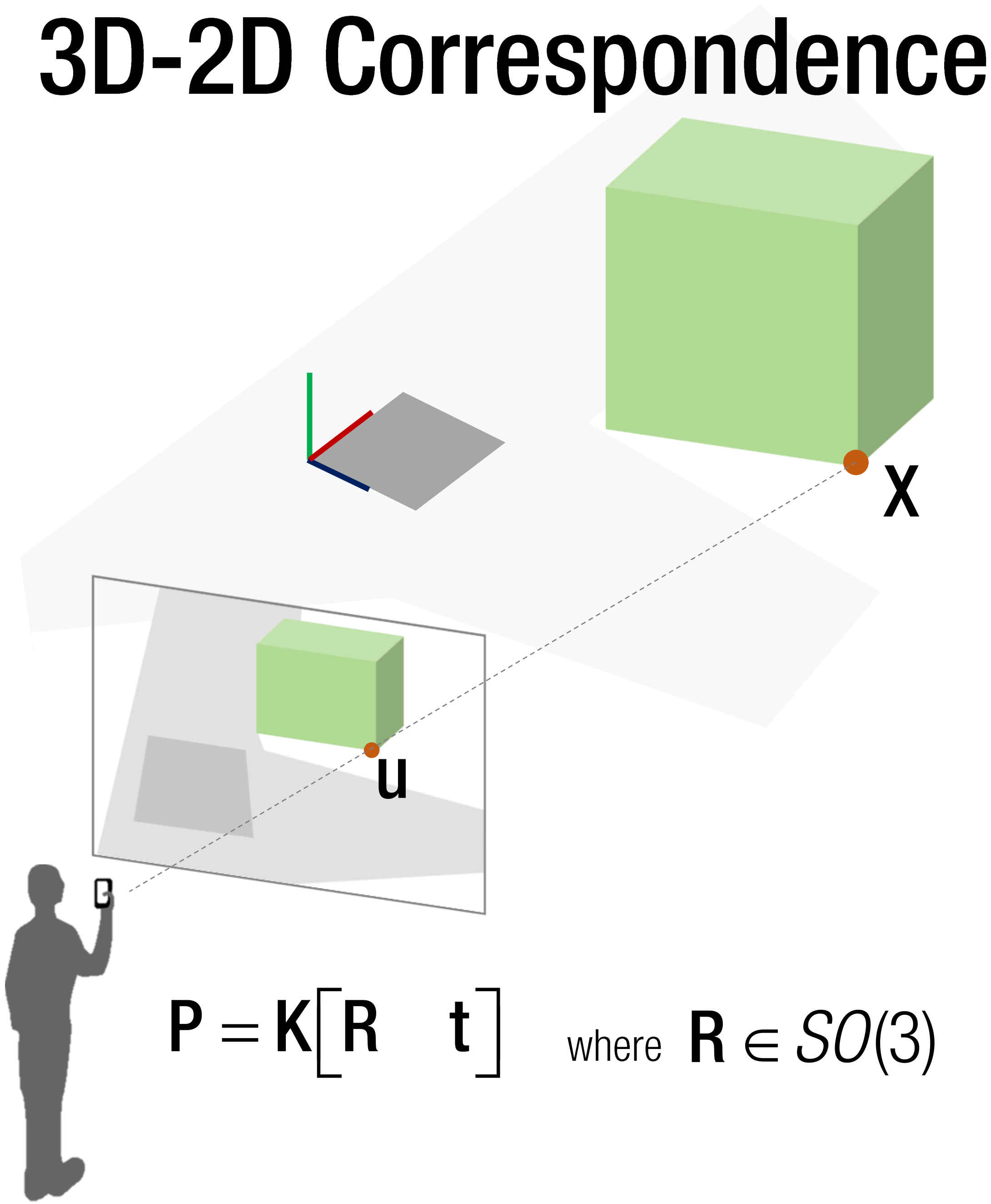
$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} \quad \text{where } \mathbf{R} \in SO(3)$$

3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

$$\lambda \mathbf{u} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

$$\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

3D-2D Correspondence



$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} \quad \text{where } \mathbf{R} \in SO(3)$$

3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

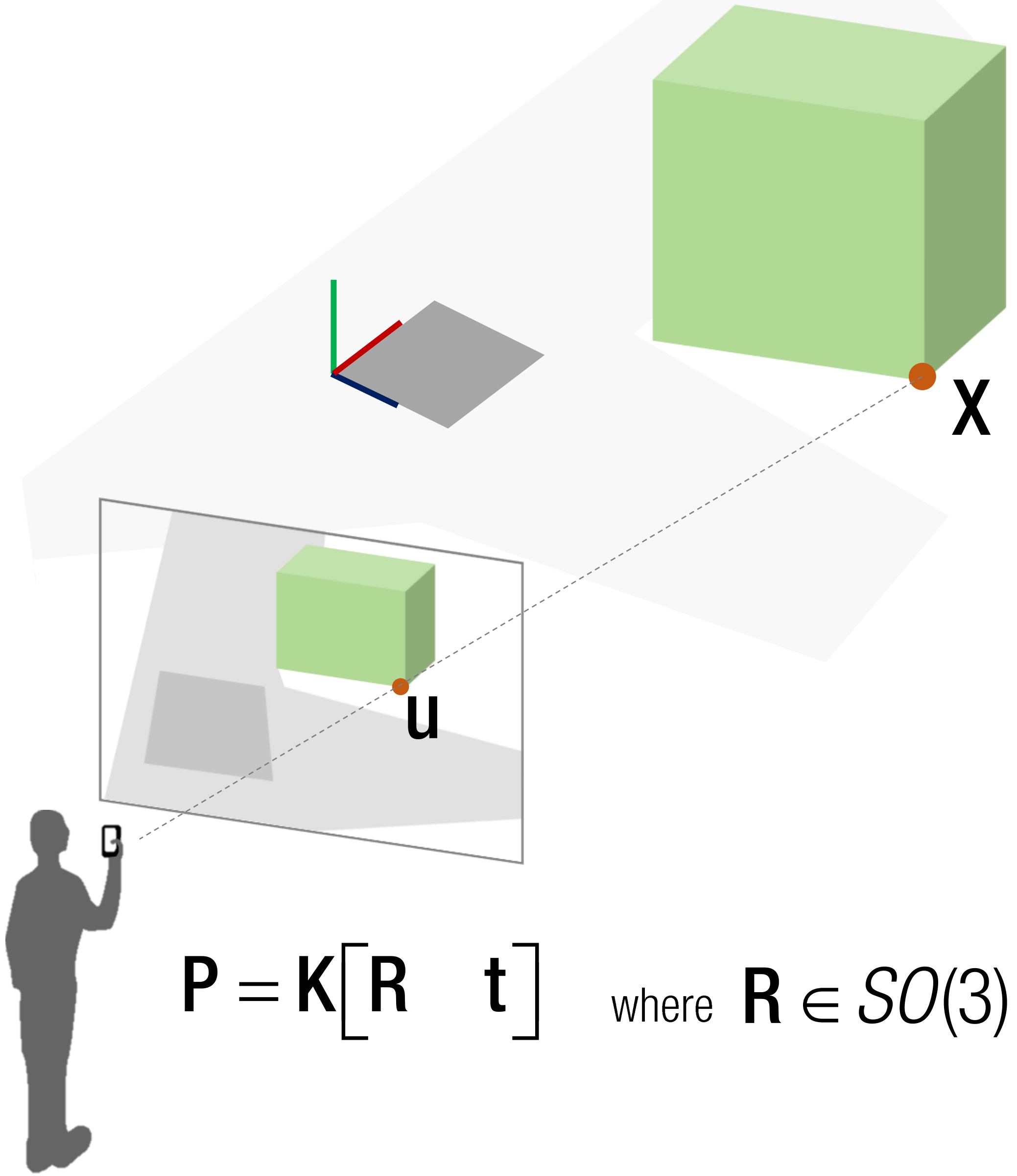
$$\lambda\mathbf{u} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

$$\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Linear in camera matrix

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}} \qquad u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

3D-2D Correspondence



$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}$$ where $\mathbf{R} \in SO(3)$

3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

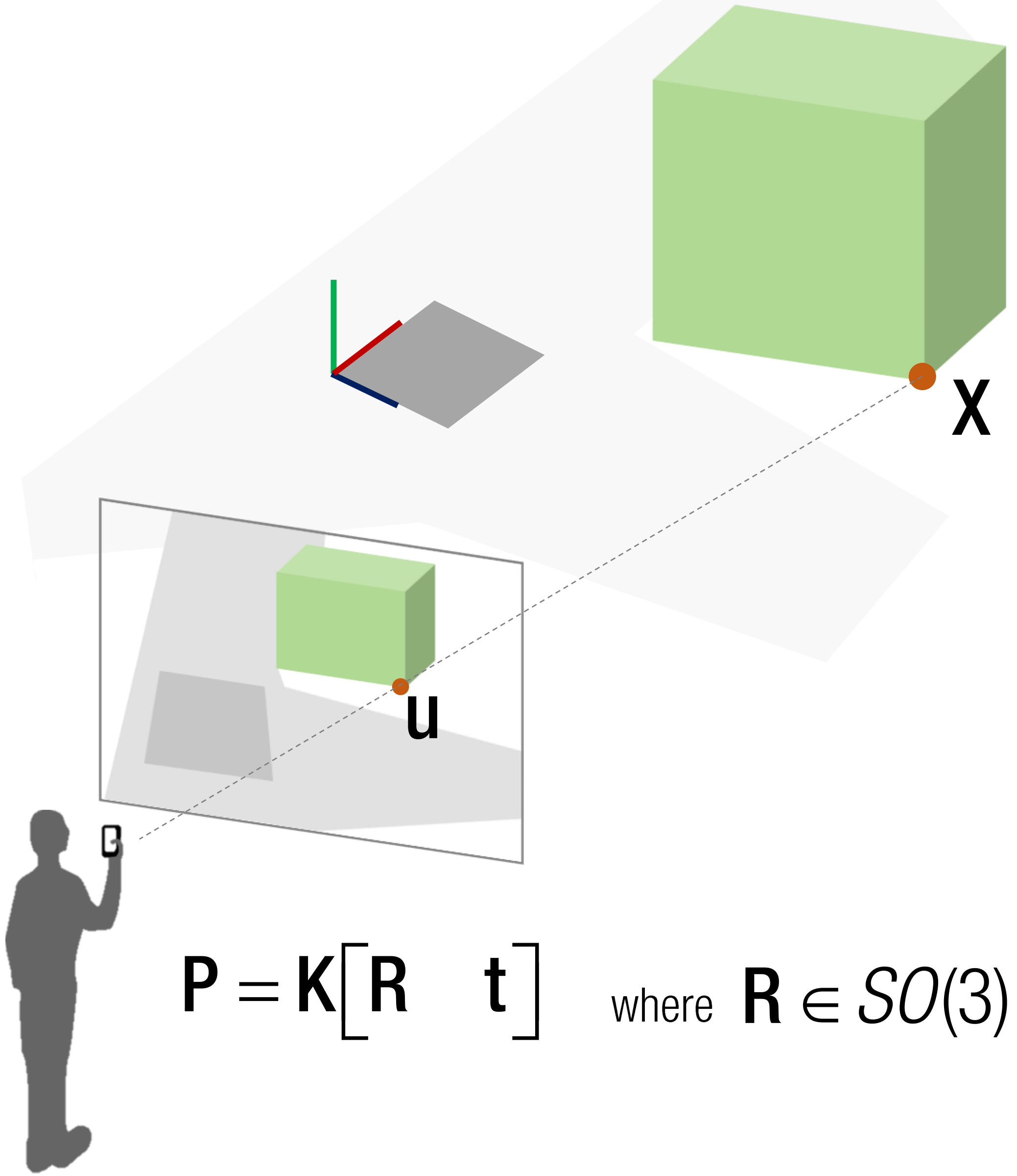
$$\lambda\mathbf{u} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

$$\underline{\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Linear in camera matrix

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}} \qquad u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

3D-2D Correspondence

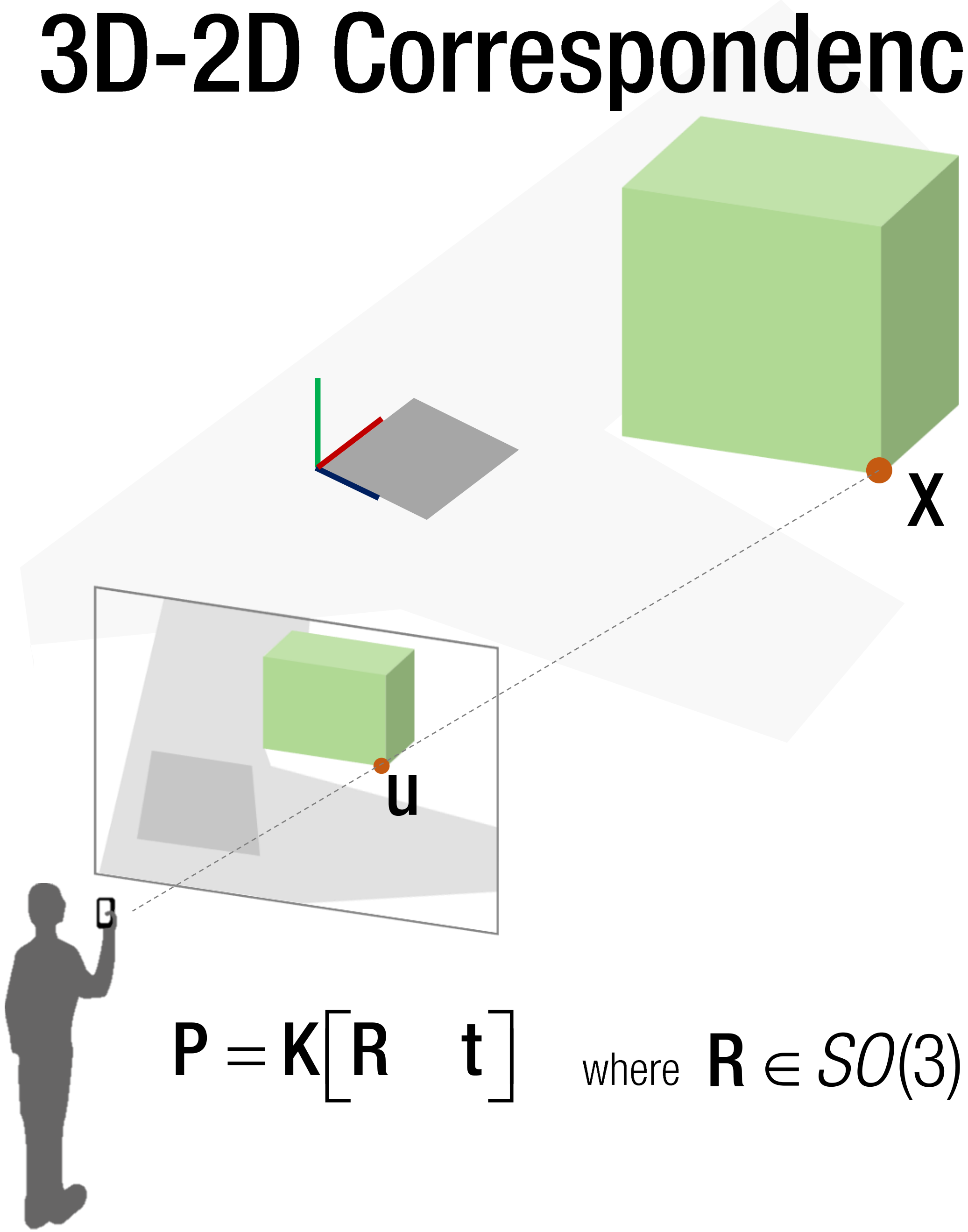


3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}} \qquad u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} \quad \text{where } \mathbf{R} \in SO(3)$$

3D-2D Correspondence



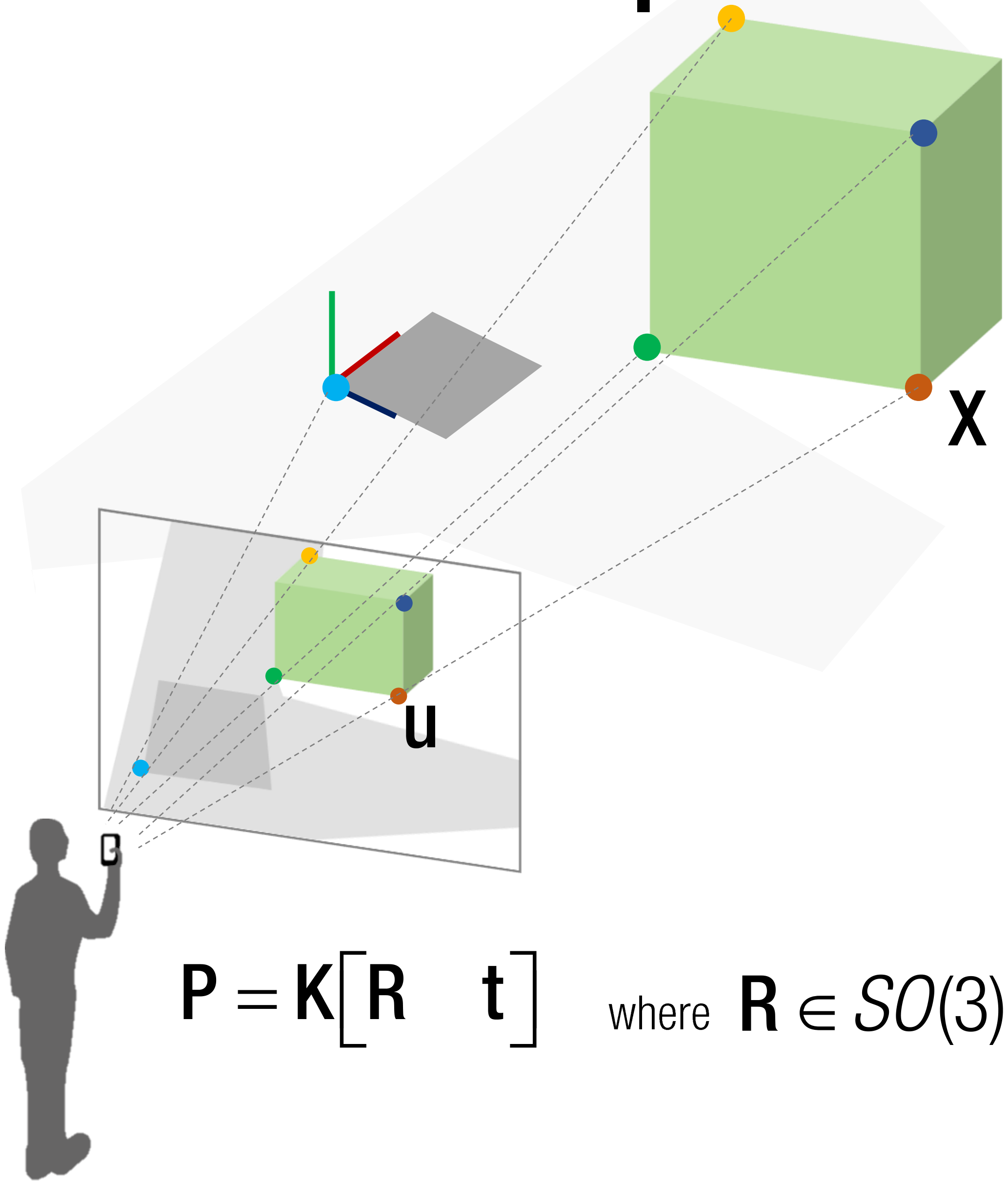
$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} \quad \text{where } \mathbf{R} \in SO(3)$$

3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}} \qquad u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$\underbrace{\begin{bmatrix} X & Y & Z & 1 & & & & & -u^xX & -u^xY & -u^xZ & -u^x \\ & & & & X & Y & Z & 1 & -u^yX & -u^yY & -u^yZ & -u^y \end{bmatrix}}_{2\text{x}12} \begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

3D-2D Correspondence



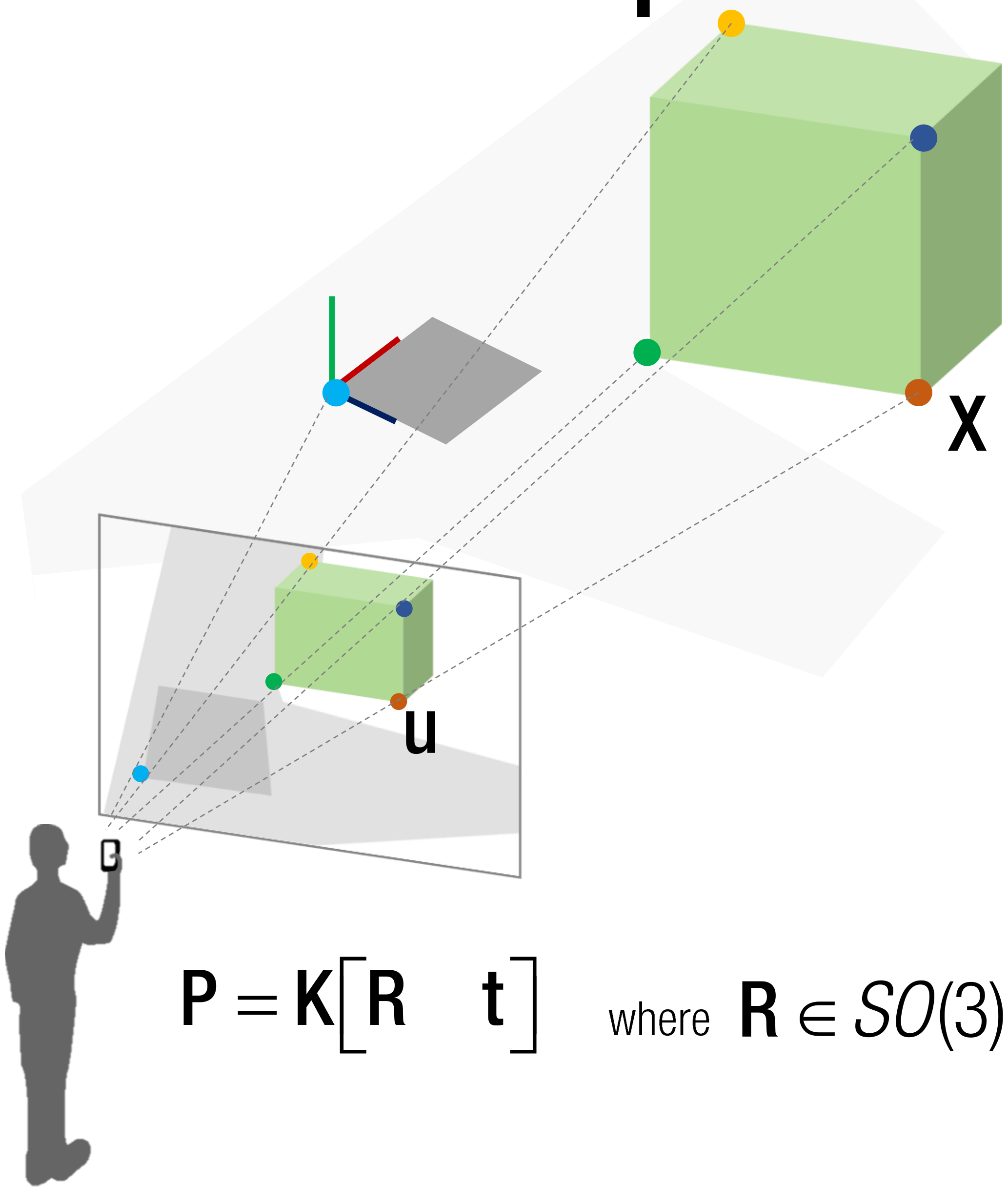
$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} \quad \text{where } \mathbf{R} \in SO(3)$$

3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}} \qquad u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$\underbrace{\begin{bmatrix}
X_1 & Y_1 & Z_1 & 1 & & & & & -u_1^x X & -u_1^x Y & -u_1^x Z & -u_1^x \\
 & & & & X_1 & Y_1 & Z_1 & 1 & -u_1^y X & -u_1^y Y & -u_1^y Z & -u_1^y \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
X_m & Y_m & Z_m & 1 & & & & & -u_m^x X & -u_m^x Y & -u_m^x Z & -u_m^x \\
 & & & & X_m & Y_m & Z_m & 1 & -u_m^y X & -u_m^y Y & -u_m^y Z & -u_m^y
\end{bmatrix}}_{2m\text{x}12}
\begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{bmatrix}
=
\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

3D-2D Correspondence



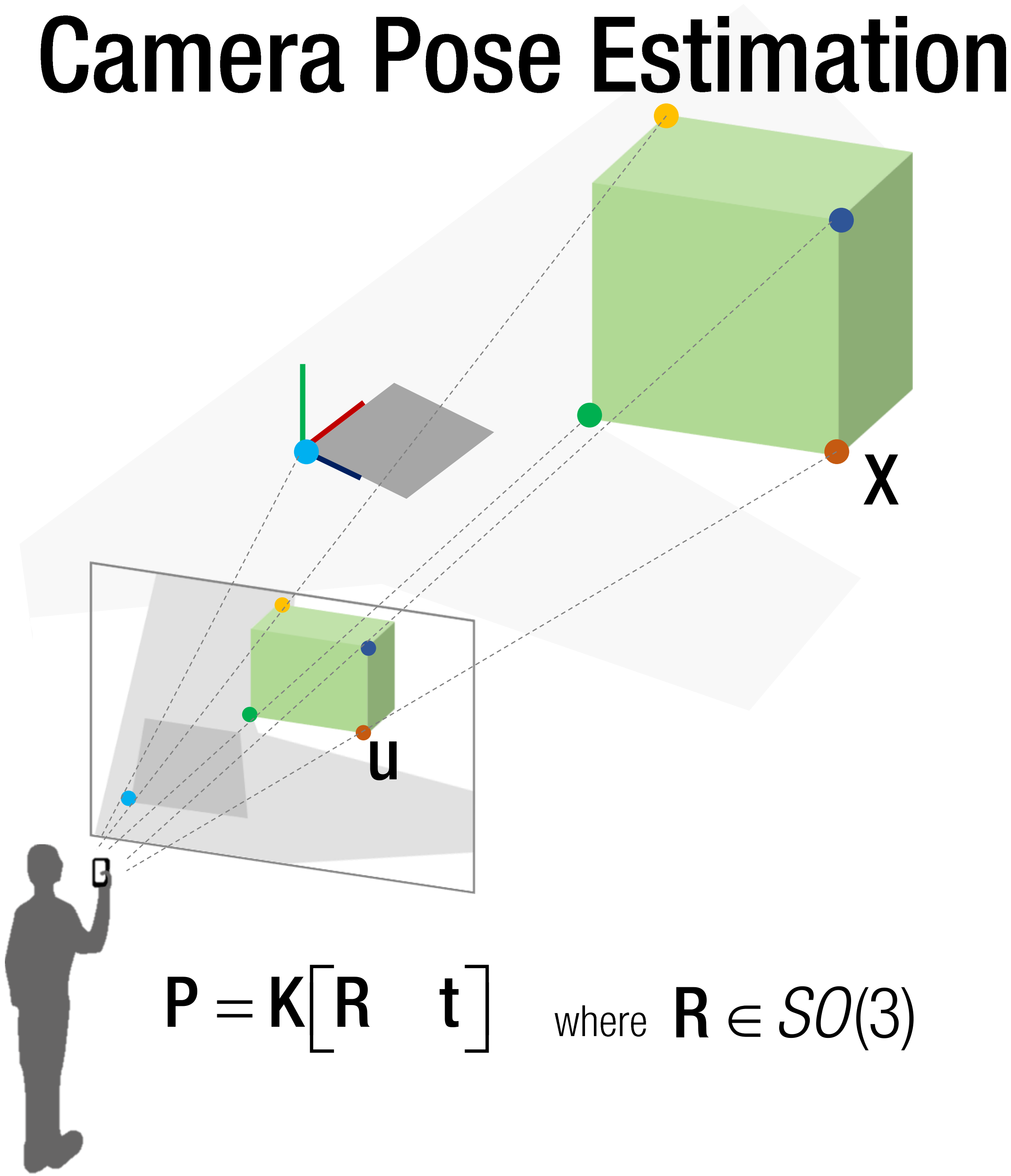
$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} \quad \text{where } \mathbf{R} \in SO(3)$$

3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}} \qquad u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$\underbrace{\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & & & & & -u_1^x X & -u_1^x Y & -u_1^x Z & -u_1^x \\ & & & & X_1 & Y_1 & Z_1 & 1 & -u_1^y X & -u_1^y Y & -u_1^y Z & -u_1^y \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ X_m & Y_m & Z_m & 1 & & & & & -u_m^x X & -u_m^x Y & -u_m^x Z & -u_m^x \\ & & & & X_m & Y_m & Z_m & 1 & -u_m^y X & -u_m^y Y & -u_m^y Z & -u_m^y \end{bmatrix}}_{\mathbf{A}\;(2m\text{x}12)} \underbrace{\begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{0}}$$

Camera Pose Estimation



$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} \quad \text{where } \mathbf{R} \in SO(3)$$

If **K** is given,

$$\mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} = \gamma\begin{bmatrix}\mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 & \mathbf{p}_4\end{bmatrix}$$

$$\longrightarrow \gamma\mathbf{R} = \mathbf{K}^{-1}\begin{bmatrix}\mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3\end{bmatrix}$$

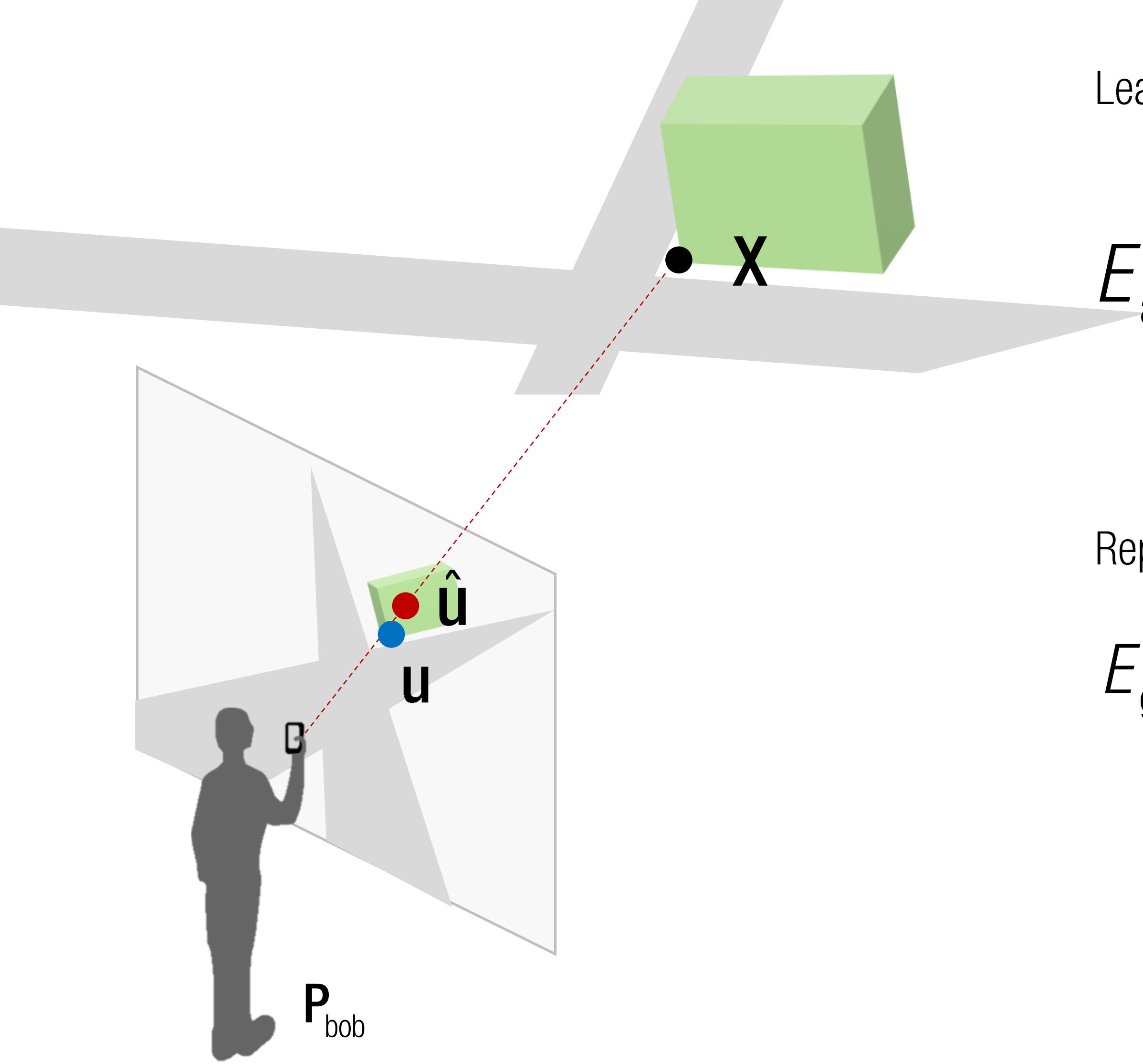
$$\mathbf{K}^{-1}\begin{bmatrix}\mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3\end{bmatrix} = \mathbf{U}\begin{bmatrix}d_{11} & & \\ & d_{22} & \\ & & d_{33}\end{bmatrix}\mathbf{V}^{\mathsf{T}}$$

$$\longrightarrow \gamma \approx d_{11}$$

$$\mathbf{R} = \mathbf{U}\mathbf{V}^{\mathsf{T}}$$: SVD cleanup

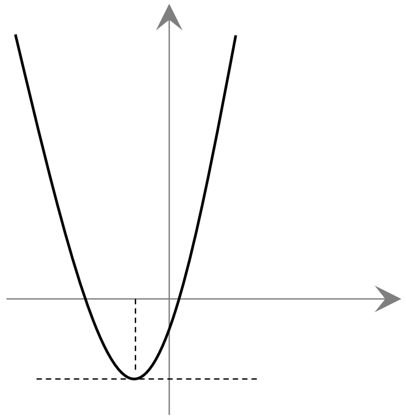
$$\longrightarrow \mathbf{t} = \frac{\mathbf{K}^{-1}\mathbf{p}_4}{d_{11}}$$: Translation and scale recovery

Algebraic vs. Geometric error



Least squares solution (algebraic error):

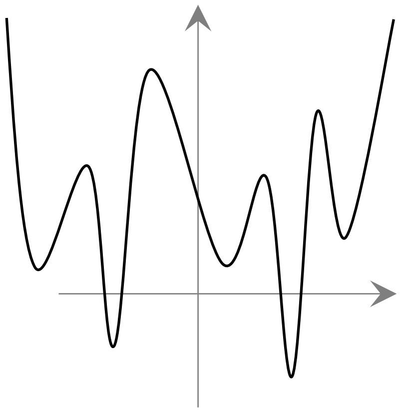
$$E_{\text{alge}} = \| \mathbf{A}\mathbf{x} - \mathbf{b} \|^2$$



Reprojection error (geometric error):

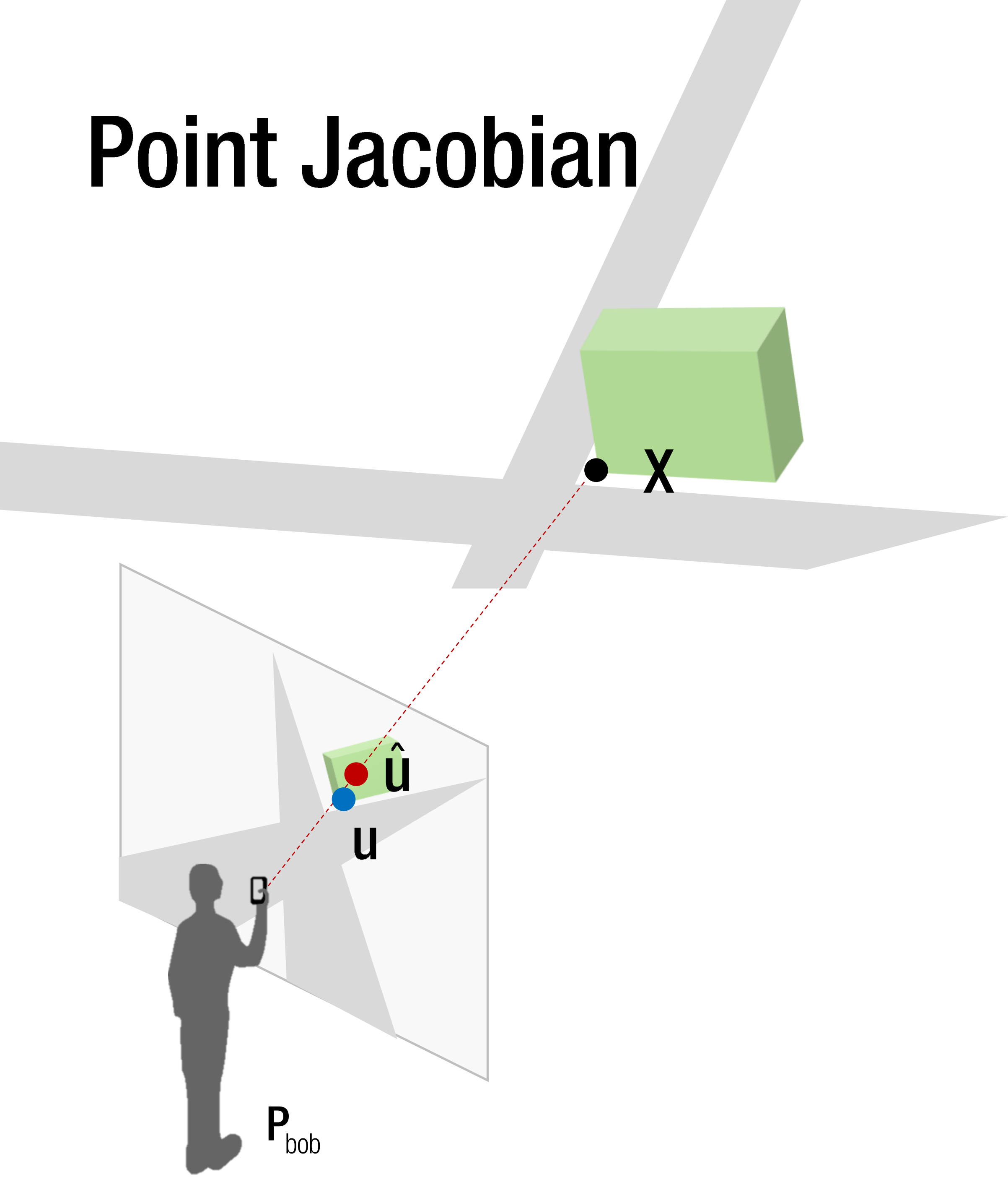
$$E_{\text{geom}} = \| \hat{\mathbf{u}} - \mathbf{u} \|^2$$

$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - \mathbf{u}_1 \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - \mathbf{u}_2 \right)^2$$



Black: given variables
Red: unknowns

Point Jacobian

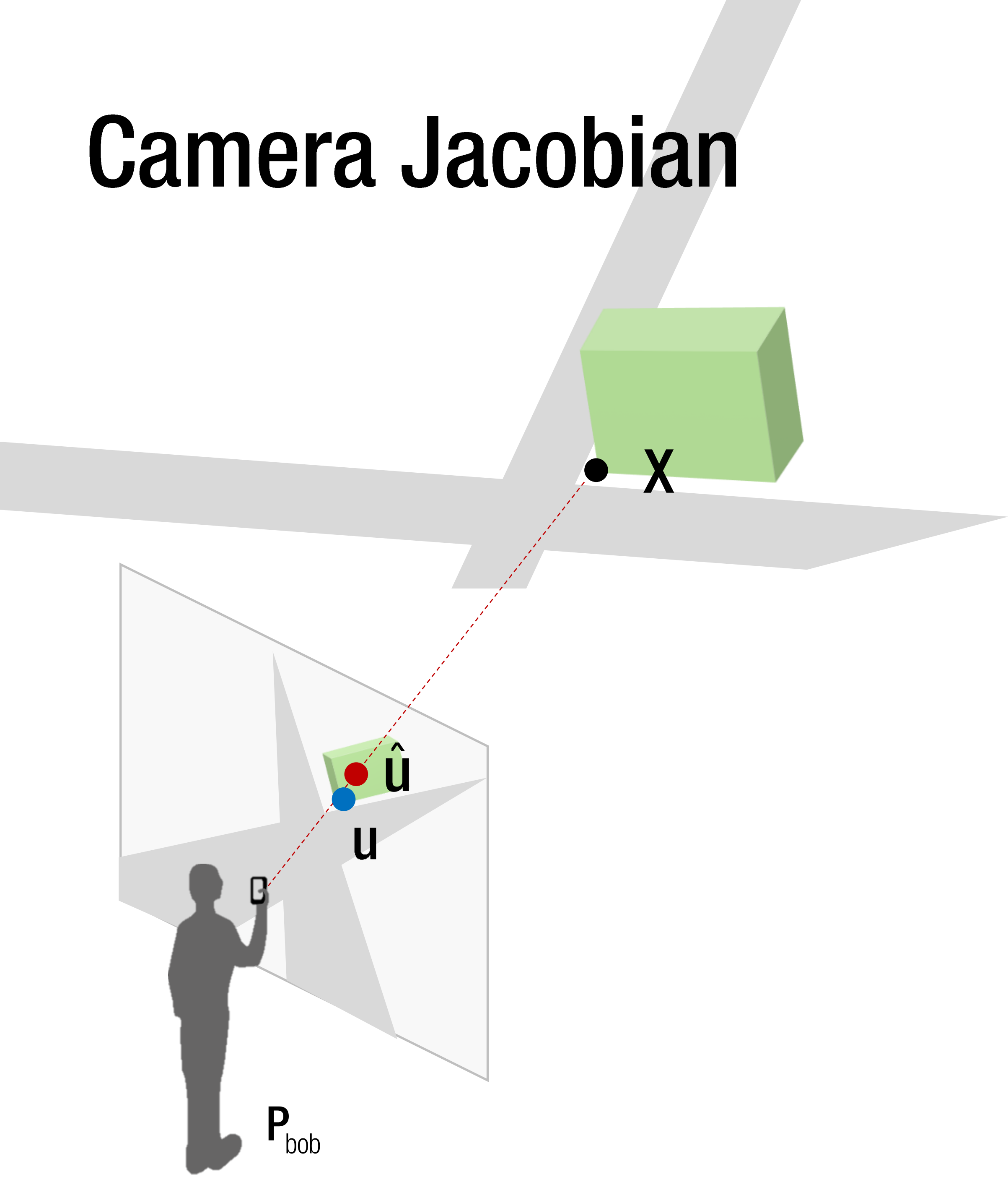


$$E_{geom} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

$$= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - y \right)^2$$

Black: given variables
Red: unknowns

Camera Jacobian

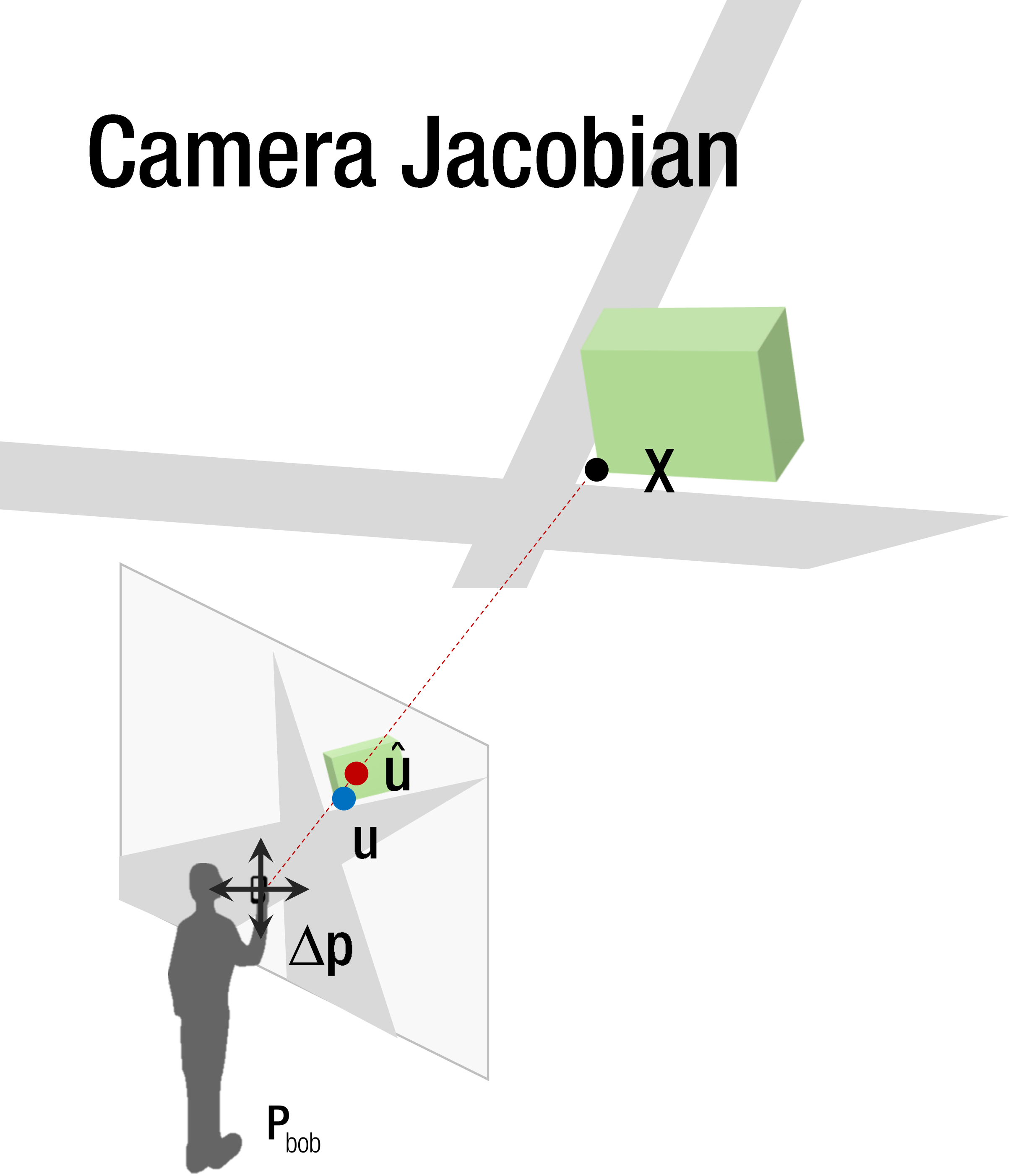


$$E_{\text{geom}} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

$$= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - y \right)^2$$

Camera Jacobian

Black: given variables
Red: unknowns



$$E_{\mathrm{geom}} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

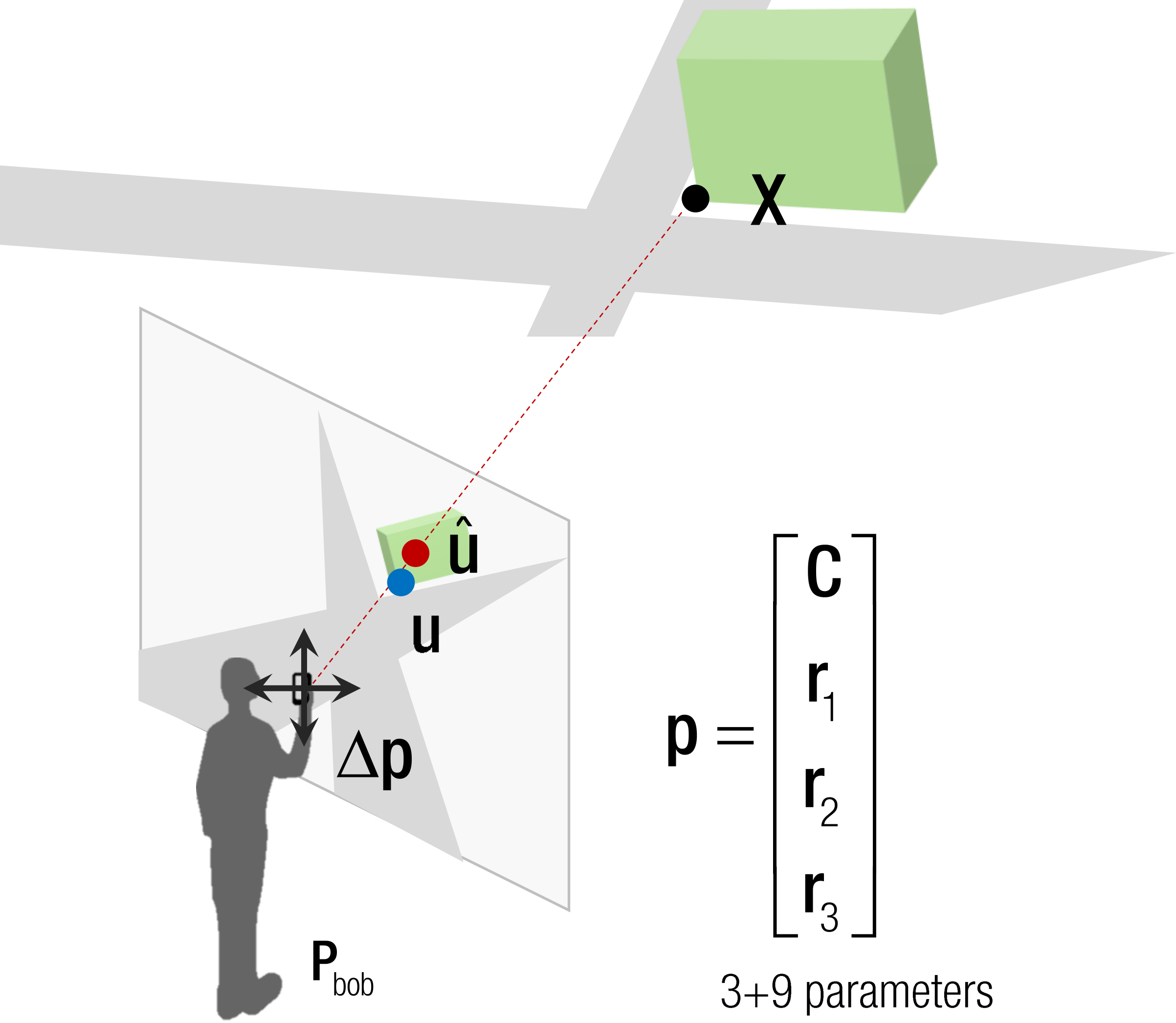
$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - y \right)^2$$

$$\Delta\mathbf{p} = \left(\frac{\partial f(\mathbf{p})^{\mathsf{T}}}{\partial \mathbf{p}} \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} \right)^{-1} \frac{\partial f(\mathbf{p})^{\mathsf{T}}}{\partial \mathbf{p}} \left(\mathbf{b} - f(\mathbf{p}) \right)$$

Camera Jacobian

Black: given variables
Red: unknowns

$$E_{\text{geom}} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X} - \mathbf{C})$$



$$\rightarrow \quad \frac{\partial}{\partial \mathbf{C}}\begin{bmatrix} u \\ v \\ w \end{bmatrix} = -\mathbf{KR}$$

$$\rightarrow \quad \frac{\partial}{\partial \mathbf{R}}\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \mathbf{K}_{11}(\mathbf{X}-\mathbf{C}) & \mathbf{0}_{1\times 3} & \mathbf{K}_{13}(\mathbf{X}-\mathbf{C}) \\ \mathbf{0}_{1\times 3} & \mathbf{K}_{22}(\mathbf{X}-\mathbf{C}) & \mathbf{K}_{23}(\mathbf{X}-\mathbf{C}) \\ \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} & (\mathbf{X}-\mathbf{C}) \end{bmatrix}$$

$$\mathbf{p} = \begin{bmatrix} \mathbf{C} \\ \mathbf{r}_1 \\ \mathbf{r}_2 \\ \mathbf{r}_3 \end{bmatrix}$$

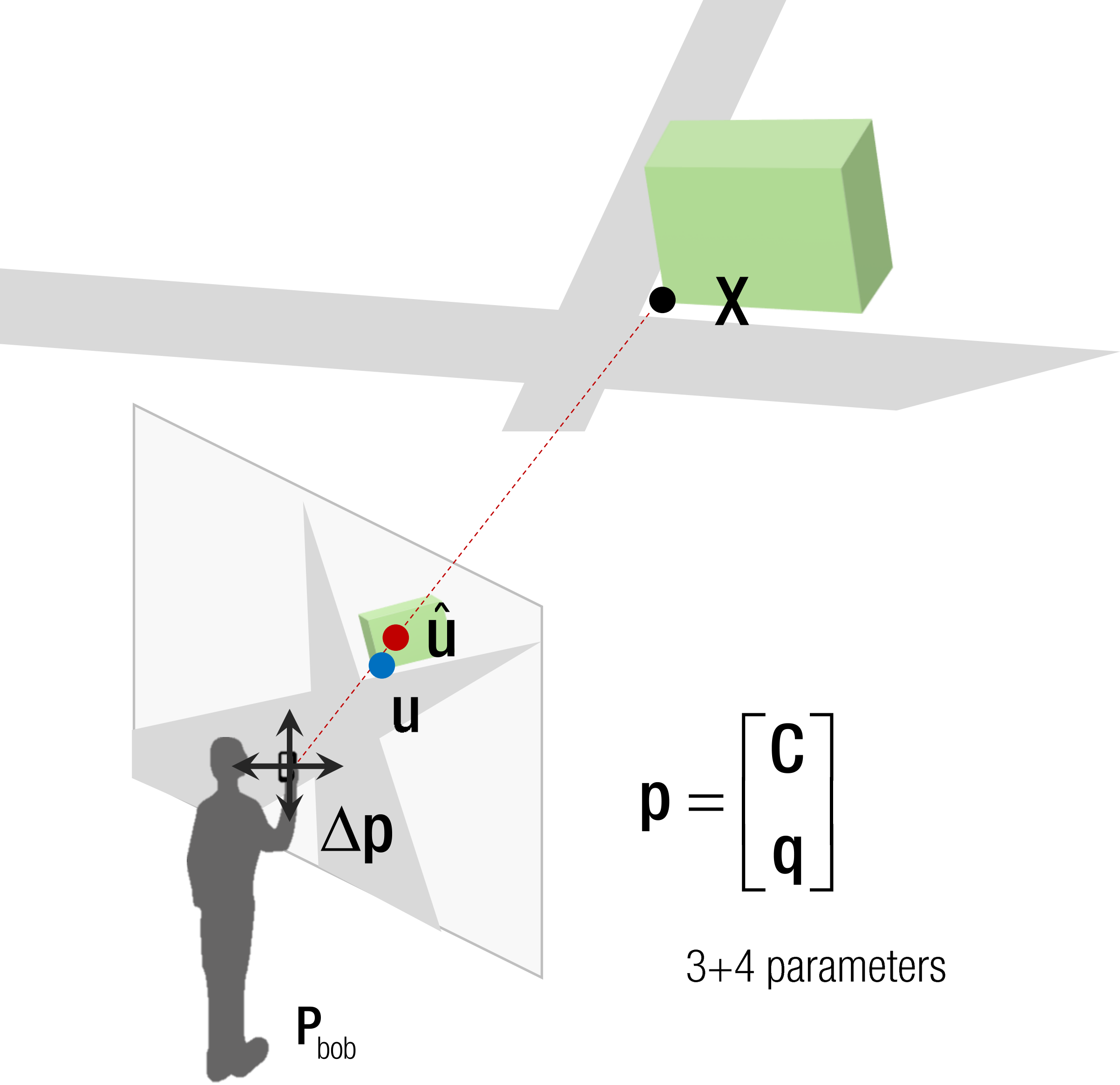
3+9 parameters

$$\Delta \mathbf{p} = \left(\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}^{\mathsf{T}} \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}\right)^{-1} \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}^{\mathsf{T}} \left(\mathbf{b} - f(\mathbf{p})\right)$$

Camera Jacobian

Black: given variables

Red: unknowns



$$\mathbf{p} = \begin{bmatrix} \mathbf{C} \\ \mathbf{q} \end{bmatrix}$$

3+4 parameters

$$E_{\text{geom}} = \left(\frac{u}{w} - x \right)^2 + \left(\frac{v}{w} - y \right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X} - \mathbf{C})$$

$$\rightarrow \quad \frac{\partial}{\partial \mathbf{C}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = -\mathbf{KR}$$

$$\rightarrow \quad \frac{\partial}{\partial \mathbf{R}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \mathbf{K}_{11}(\mathbf{X}-\mathbf{C}) & \mathbf{0}_{1\times3} & \mathbf{K}_{13}(\mathbf{X}-\mathbf{C}) \\ \mathbf{0}_{1\times3} & \mathbf{K}_{22}(\mathbf{X}-\mathbf{C}) & \mathbf{K}_{23}(\mathbf{X}-\mathbf{C}) \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & (\mathbf{X}-\mathbf{C}) \end{bmatrix}$$

$$\rightarrow \quad \frac{\partial}{\partial \mathbf{q}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \frac{\partial}{\partial \mathbf{R}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \underbrace{\frac{\partial \mathbf{R}}{\partial \mathbf{q}}}_{\text{Quaternion jacobian}}$$

: Chain rule

Quaternion Jacobian

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

where

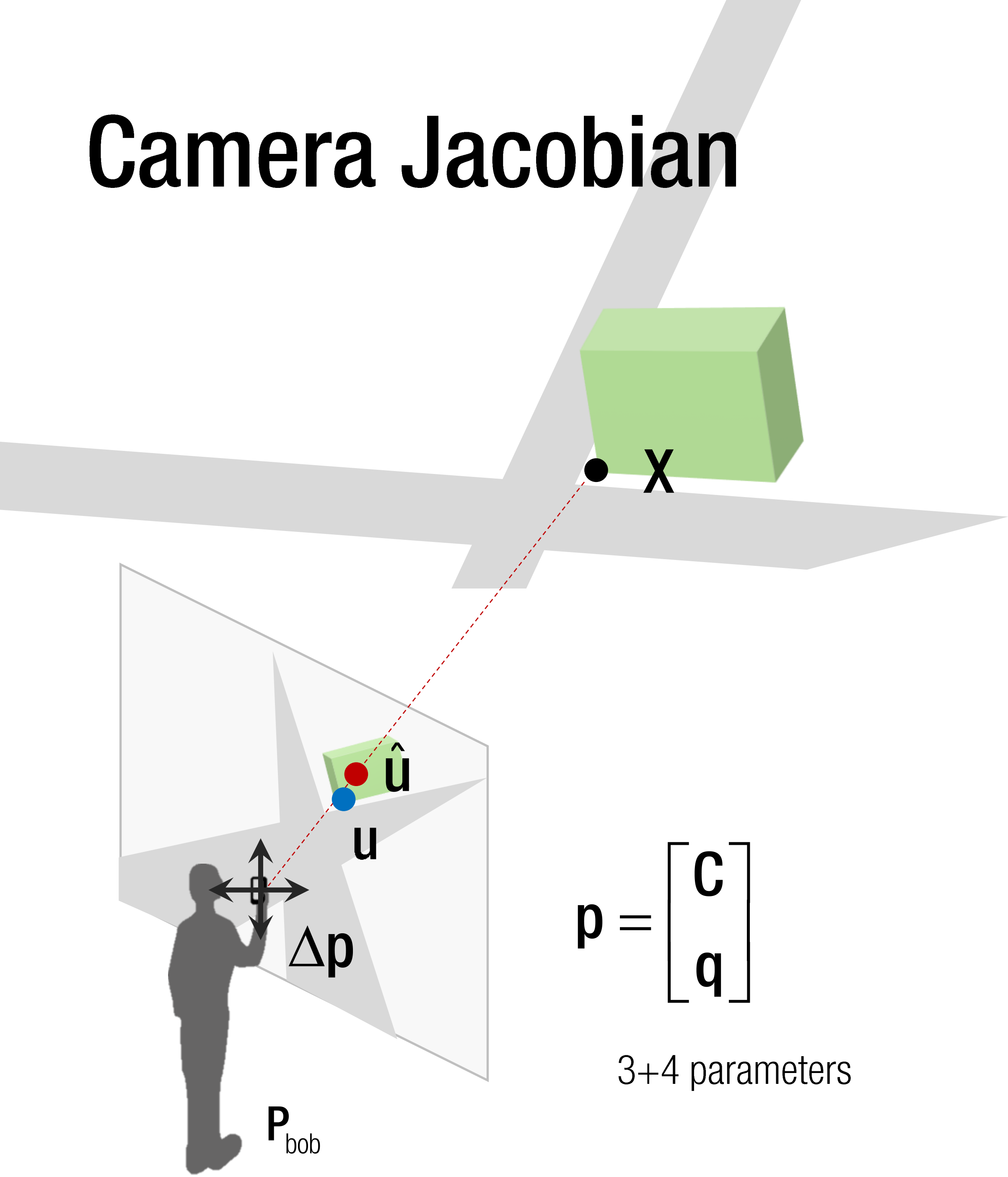
$$\mathbf{q} = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix}^\mathsf{T}$$

$$\underbrace{\frac{\partial \mathbf{R}}{\partial \mathbf{q}}}_{9\mathrm{x}4} = \begin{bmatrix} 0 & 0 & -4q_y & -4q_z \\ -2q_z & 2q_y & 2q_x & -2q_w \\ 2q_y & 2q_z & 2q_w & 2q_x \\ 2q_z & 2q_y & 2q_x & 2q_w \\ 0 & -4q_x & 0 & -4q_z \\ -2q_x & -2q_w & 2q_z & 2q_y \\ -2q_y & 2q_z & -2q_w & 2q_x \\ 2q_x & 2q_w & 2q_z & 2q_y \\ 0 & -4q_x & -4q_y & 0 \end{bmatrix}$$

Camera Jacobian

Black: given variables
Red: unknowns

$$E_{geom} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X}-\mathbf{C})$$



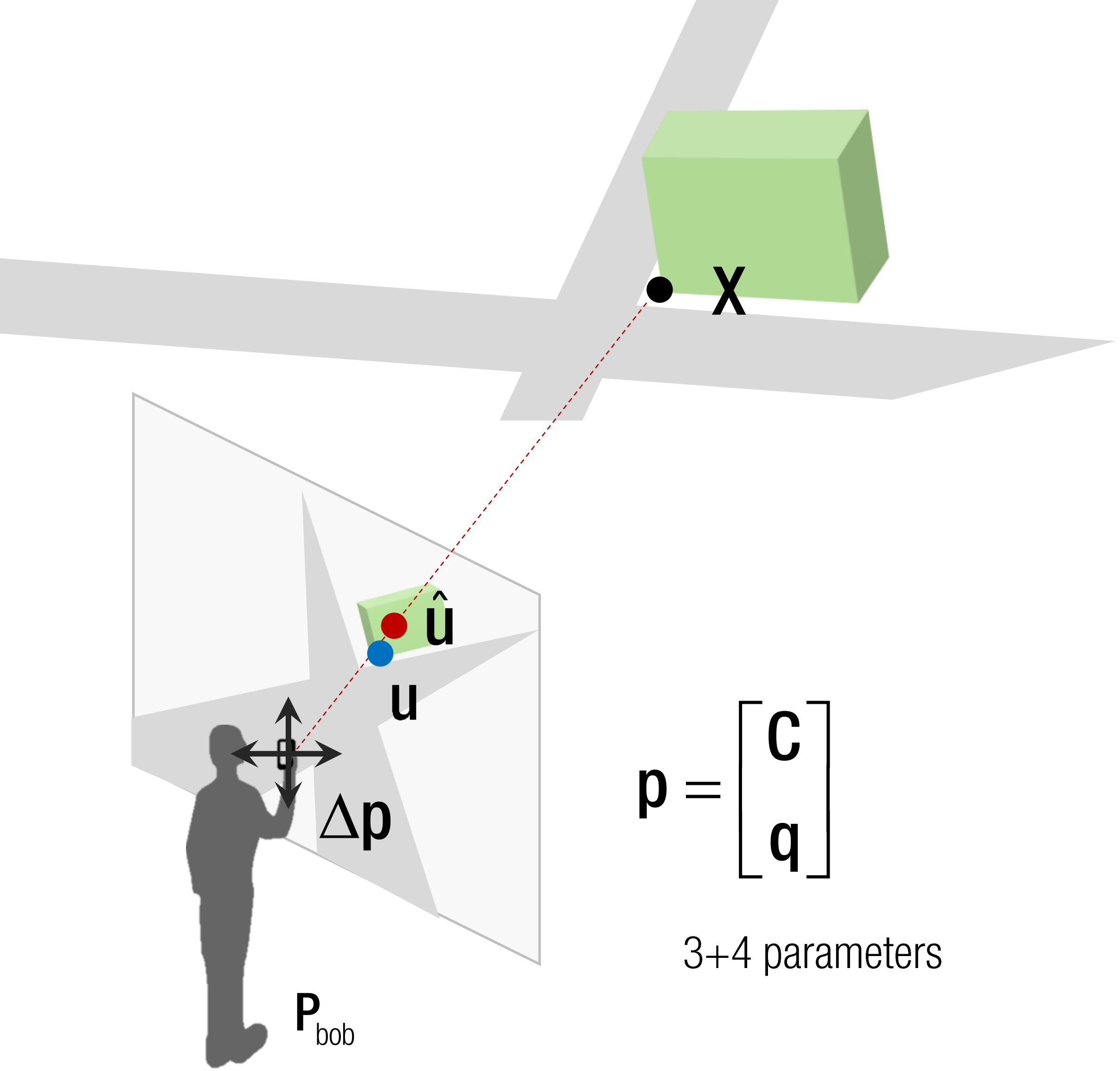
$$\mathbf{p} = \begin{bmatrix} \mathbf{C} \\ \mathbf{q} \end{bmatrix}$$

3+4 parameters

$$f(\mathbf{p}) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} \longrightarrow \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\frac{\partial u}{\partial \mathbf{p}} - u\frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v\frac{\partial u}{\partial \mathbf{p}} - v\frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

Camera Jacobian

Black: given variables
Red: unknowns



$$E_{\text{geom}} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X} - \mathbf{C})$$

$$\longrightarrow \quad \frac{\partial}{\partial \mathbf{C}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = -\mathbf{KR} \qquad \frac{\partial}{\partial \mathbf{q}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \frac{\partial}{\partial \mathbf{R}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \frac{\partial \mathbf{R}}{\partial \mathbf{q}}$$

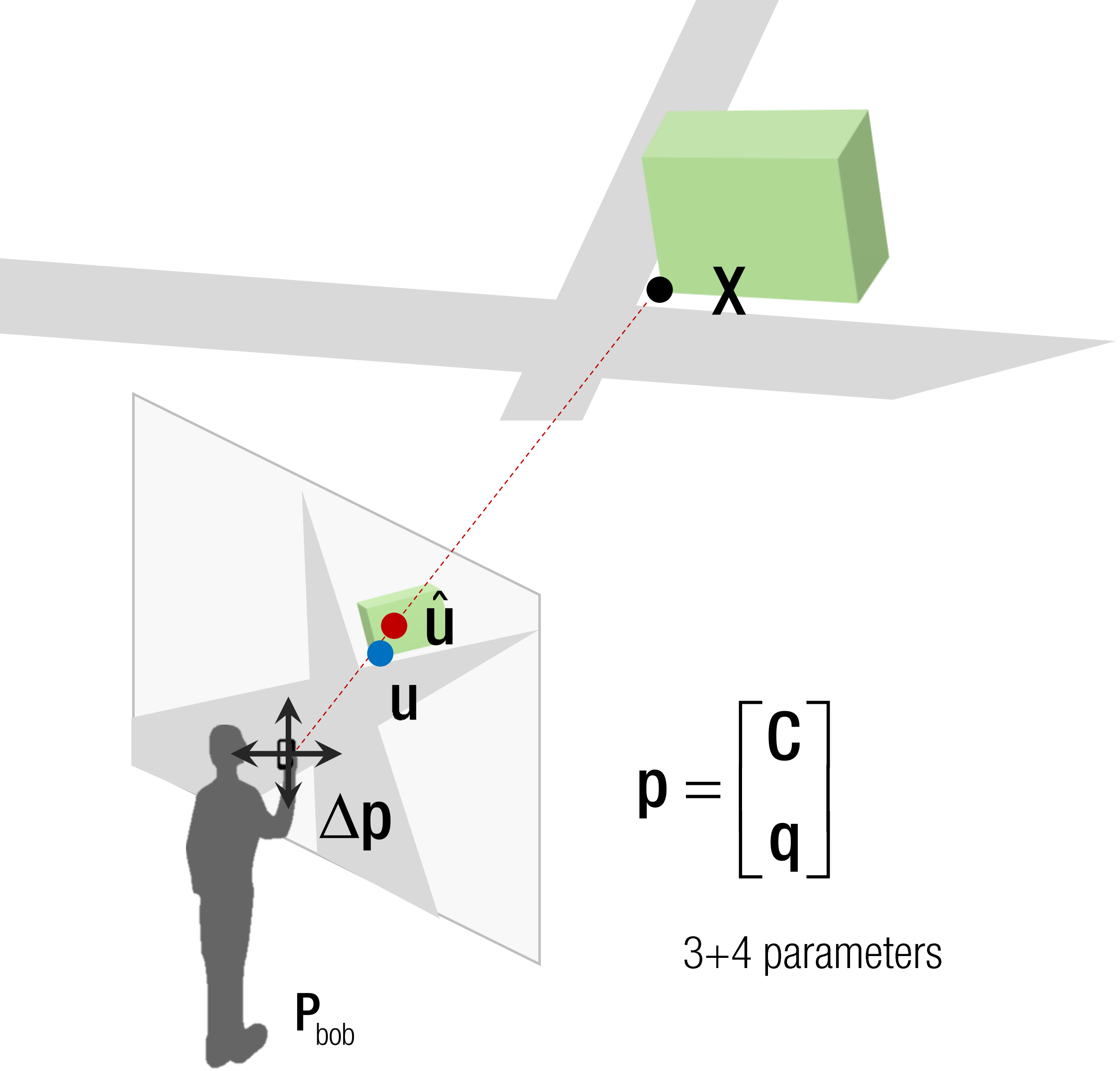
$$\mathbf{p} = \begin{bmatrix} \mathbf{C} \\ \mathbf{q} \end{bmatrix}$$

3+4 parameters

$$f(\mathbf{p}) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} \quad \longrightarrow \quad \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w \frac{\partial u}{\partial \mathbf{p}} - u \frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v \frac{\partial u}{\partial \mathbf{p}} - v \frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

Camera Jacobian

Black: given variables
Red: unknowns



$$\mathbf{p} = \begin{bmatrix} \mathbf{C} \\ \mathbf{q} \end{bmatrix}$$

3+4 parameters

$$E_{\text{geom}} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{KR}(\mathbf{X} - \mathbf{C})$$

$$\rightarrow \quad \frac{\partial}{\partial \mathbf{C}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = -\mathbf{KR} \qquad \frac{\partial}{\partial \mathbf{q}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \frac{\partial}{\partial \mathbf{R}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \frac{\partial \mathbf{R}}{\partial \mathbf{q}}$$

$$\rightarrow \quad \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial \mathbf{C}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} & \frac{\partial}{\partial \mathbf{q}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \end{bmatrix}$$

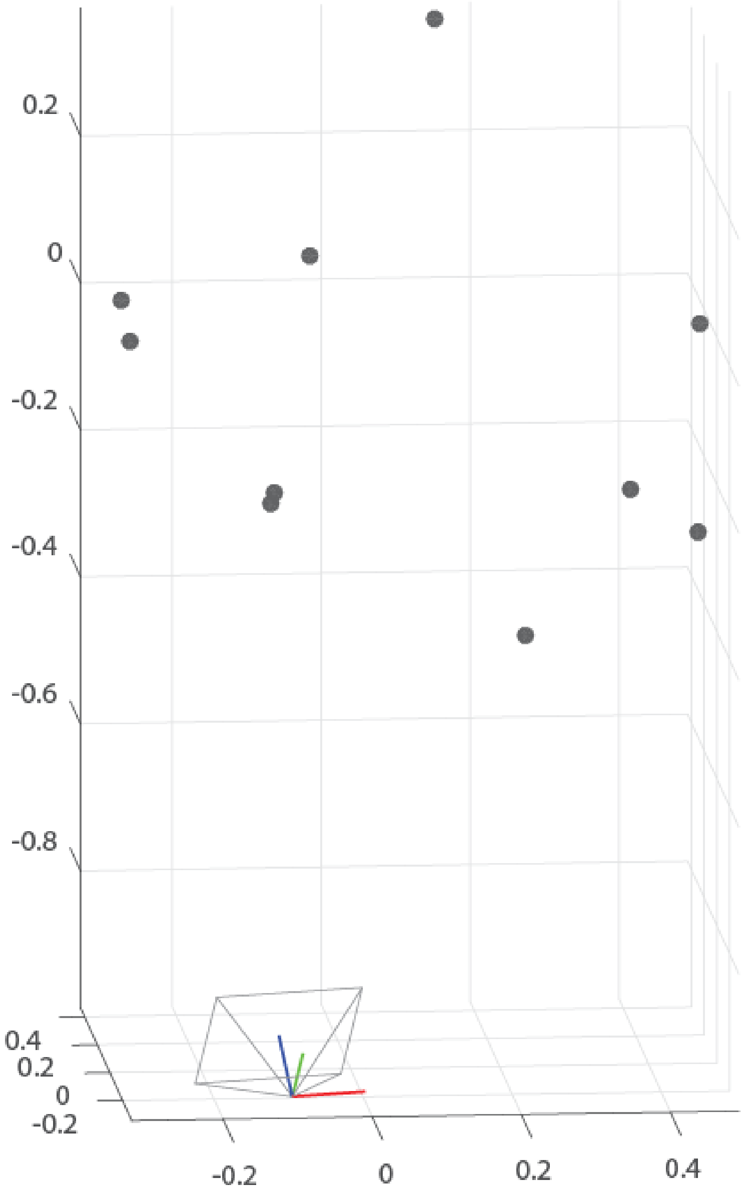
$$f(\mathbf{p}) = \begin{bmatrix} \dfrac{u}{w} \\ \dfrac{v}{w} \end{bmatrix} \quad \rightarrow \quad \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} \dfrac{u}{w} \\ \dfrac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w \frac{\partial u}{\partial \mathbf{p}} - u \frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v \frac{\partial u}{\partial \mathbf{p}} - v \frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

Algorithm 2 Nonlinear Camera Pose Refinement

$\mathbf{p} = [\mathbf{C}^\mathsf{T}\mathbf{q}^\mathsf{T}]^\mathsf{T}$
for j = 1 : nIters **do**
  $\mathbf{C} = \mathbf{p}_{1:3}$, $\mathbf{R}$=`Quaternion2Rotation`$(\mathbf{q})$, $\mathbf{q} = \mathbf{p}_{4:7}$
  Build camera pose Jacobian for all points, $\frac{\partial f(\mathbf{p})_j}{\partial \mathbf{p}} = \begin{bmatrix} \frac{\partial f(\mathbf{p})_j}{\partial \mathbf{C}} & \frac{\partial f(\mathbf{p})_j}{\partial \mathbf{q}} \end{bmatrix}$.
  Compute $f(\mathbf{p})$.
  $\Delta\mathbf{p} = \left(\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}^\mathsf{T} \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} + \lambda\mathbf{I}\right)^{-1} \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}^\mathsf{T} (\mathbf{b} - f(\mathbf{p}))$ using Equation (2).
  $\mathbf{p} = \mathbf{p} + \Delta\mathbf{p}$
  Normalize the quaternion scale, $\mathbf{p}_{4:7} = \mathbf{p}_{4:7}/\|\mathbf{p}_{4:7}\|$.
end for

$$\Delta\mathbf{p} = \left(\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}^\mathsf{T} \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} + \lambda\mathbf{I}\right)^{-1} \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}^\mathsf{T} (\mathbf{b} - f(\mathbf{p}))$$

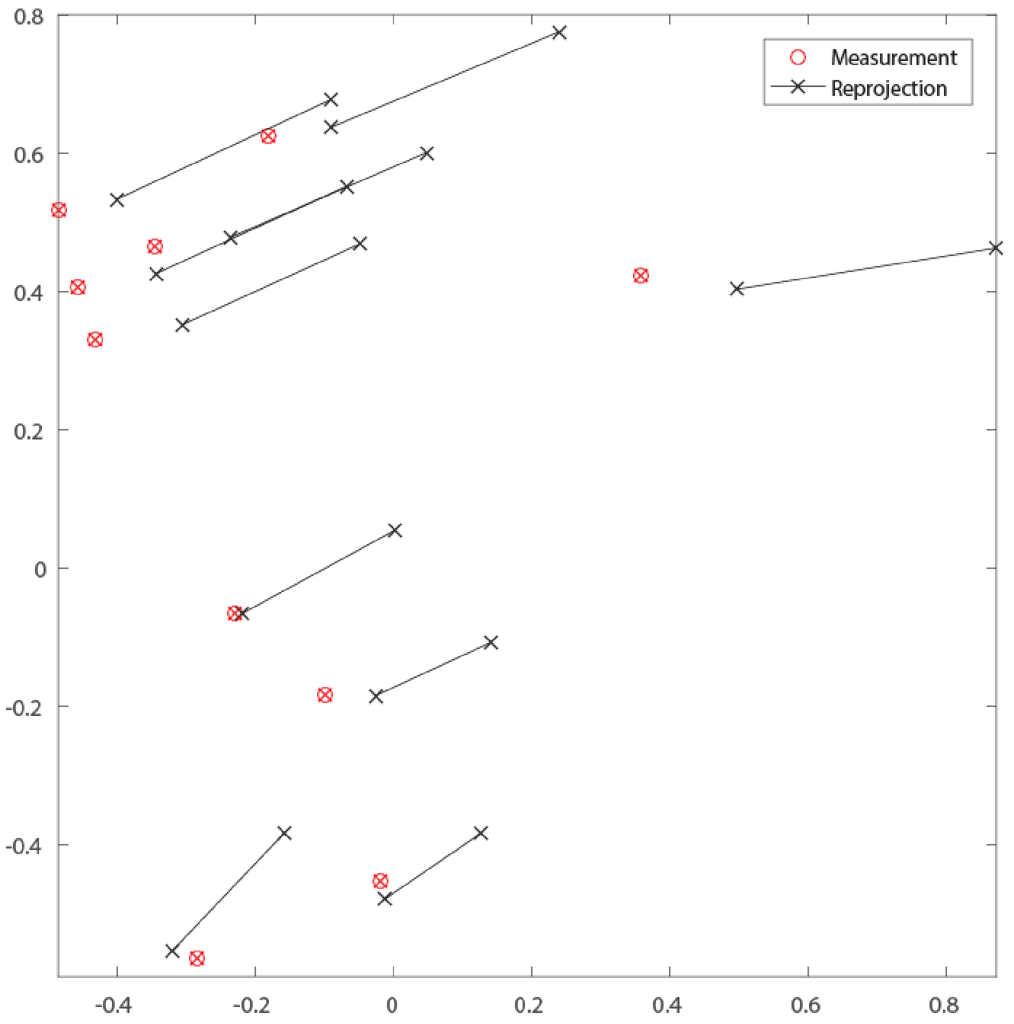
Example




```
u = K*R*[eye(3) -C]*[X'; ones(1,nPoints)];
u = [u(1,:)./u(3,:); u(2,:)./u(3,:)];

x = [C; q];
for j = 1 : 40
    R1 = Quaternion2Rotation(x(4:7));
    C1 = x(1:3);

    df_dc = [];
    df_dR = [];
    for k = 1 : nPoints
        df_dc = [df_dc; JacobianC(K, R1, C1, X(k,:)')];
        df_dR = [df_dR; JacobianR(K, R1, C1, X(k,:)')*JacobianQ(x(4:7))];
    end

    u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
    u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];

    jacobian = [df_dc df_dR];
    delta_b = u(:)-u1(:);

    delta_x = inv(jacobian'*jacobian+lambda*eye(size(jacobian'*jacobian,1))*jacobian'*delta_b
    x = x + delta_x;
    x(4:7) = x(4:7)/norm(x(4:7));
end
```

$$\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}}\begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\frac{\partial u}{\partial \mathbf{p}} - u\frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v\frac{\partial u}{\partial \mathbf{p}} - v\frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

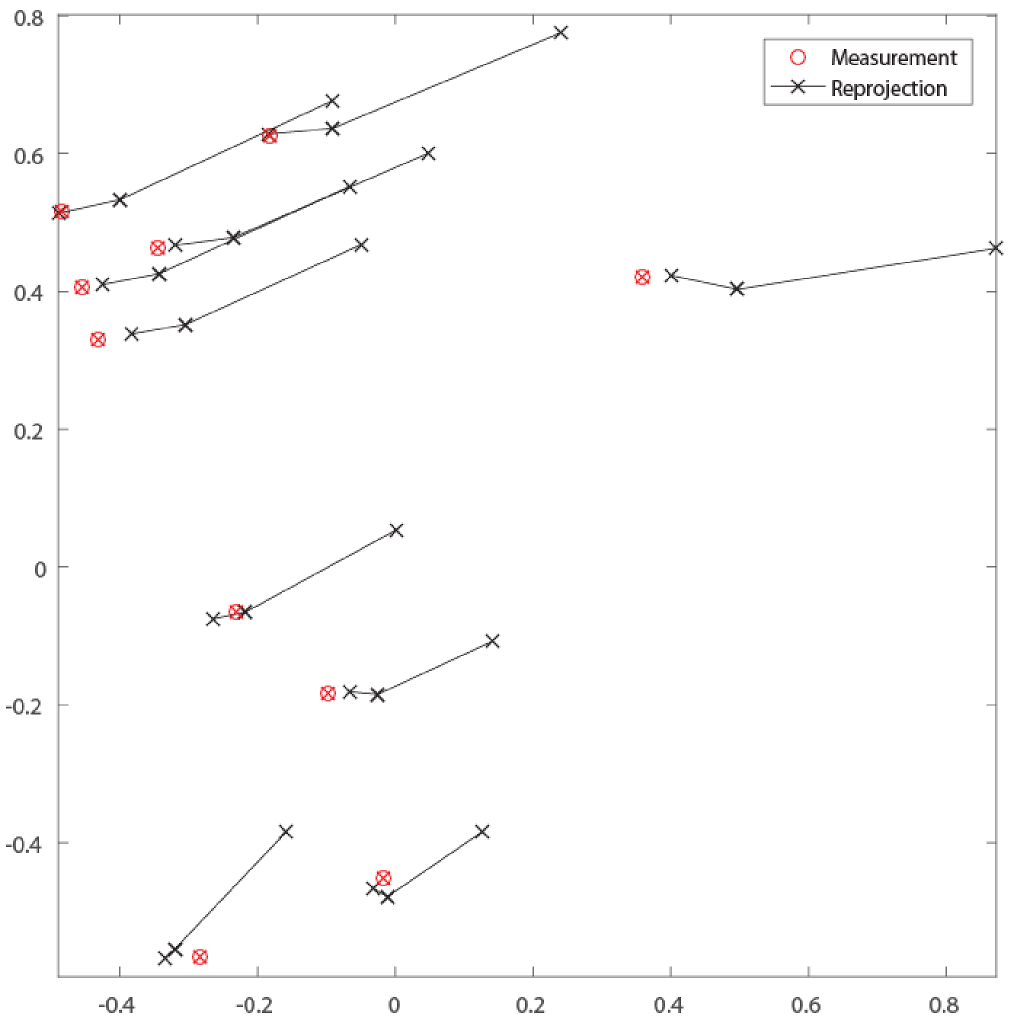
Example




```
u = K*R*[eye(3) -C]*[X'; ones(1,nPoints)];
u = [u(1,:)./u(3,:); u(2,:)./u(3,:)];

x = [C; q];
for j = 1 : 40
    R1 = Quaternion2Rotation(x(4:7));
    C1 = x(1:3);

    df_dc = [];
    df_dR = [];
    for k = 1 : nPoints
        df_dc = [df_dc; JacobianC(K, R1, C1, X(k,:)')];
        df_dR = [df_dR; JacobianR(K, R1, C1, X(k,:)')*JacobianQ(x(4:7))];
    end

    u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
    u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];

    jacobian = [df_dc df_dR];
    delta_b = u(:)-u1(:);

    delta_x = inv(jacobian'*jacobian+lambda*eye(size(jacobian'*jacobian,1))*jacobian'*delta_b
    x = x + delta_x;
    x(4:7) = x(4:7)/norm(x(4:7));
end
```

$$\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}}\begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\frac{\partial u}{\partial \mathbf{p}} - u\frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v\frac{\partial u}{\partial \mathbf{p}} - v\frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

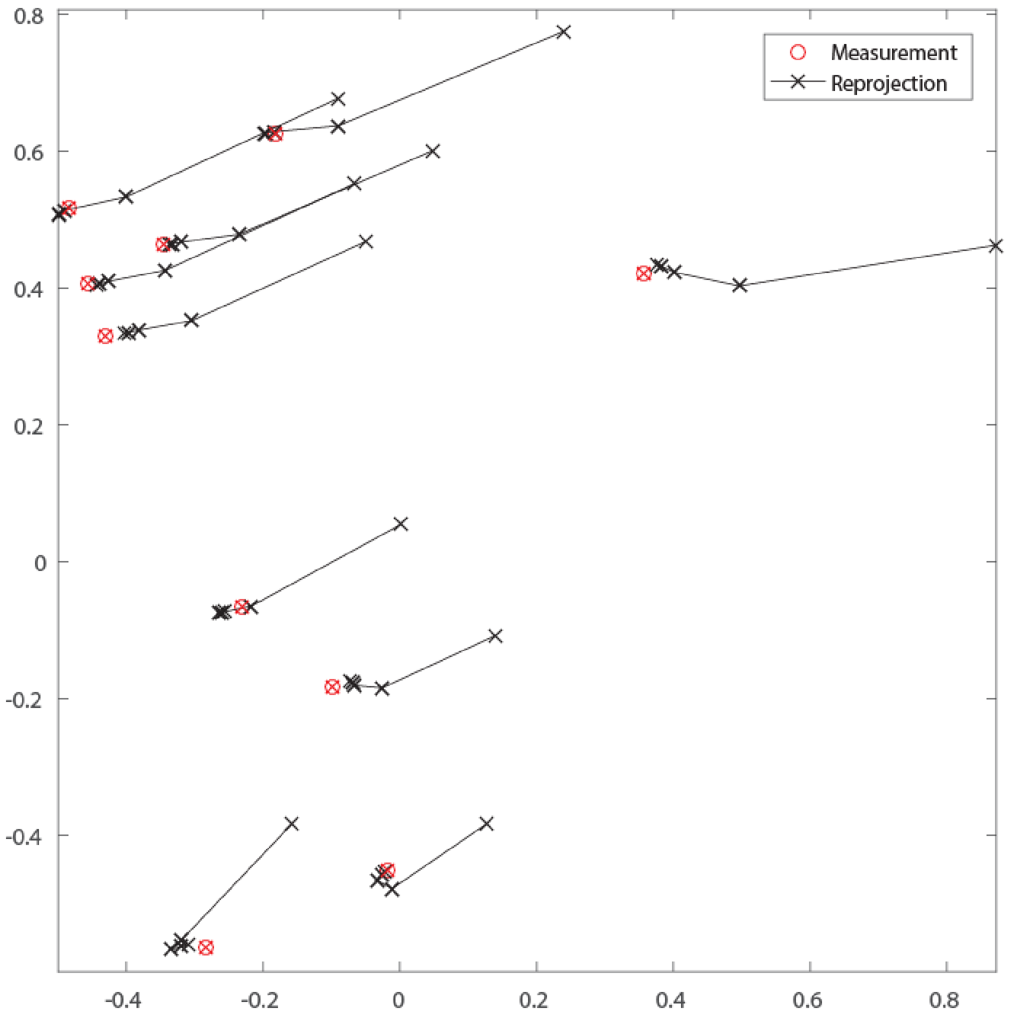
Example




```
u = K*R*[eye(3) -C]*[X'; ones(1,nPoints)];
u = [u(1,:)./u(3,:); u(2,:)./u(3,:)];

x = [C; q];
for j = 1 : 40
    R1 = Quaternion2Rotation(x(4:7));
    C1 = x(1:3);

    df_dc = [];
    df_dR = [];
    for k = 1 : nPoints
        df_dc = [df_dc; JacobianC(K, R1, C1, X(k,:)')];
        df_dR = [df_dR; JacobianR(K, R1, C1, X(k,:)')*JacobianQ(x(4:7))];
    end

    u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
    u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];

    jacobian = [df_dc df_dR];
    delta_b = u(:)-u1(:);

    delta_x = inv(jacobian'*jacobian+lambda*eye(size(jacobian'*jacobian,1))*jacobian'*delta_b
    x = x + delta_x;
    x(4:7) = x(4:7)/norm(x(4:7));
end
```

$$\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}}\begin{bmatrix} \dfrac{u}{w} \\ \dfrac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\dfrac{\partial u}{\partial \mathbf{p}} - u\dfrac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v\dfrac{\partial u}{\partial \mathbf{p}} - v\dfrac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

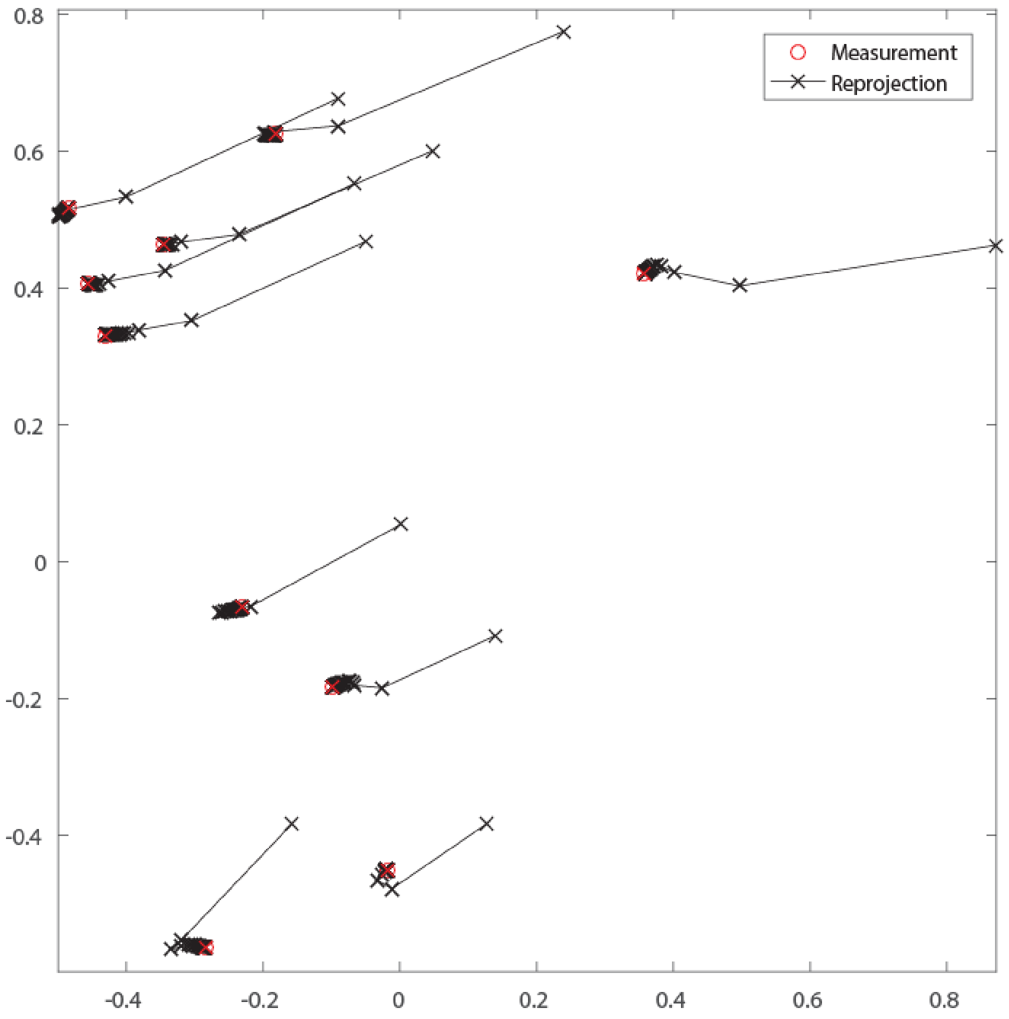
Example




```
u = K*R*[eye(3) -C]*[X'; ones(1,nPoints)];
u = [u(1,:)./u(3,:); u(2,:)./u(3,:)];

x = [C; q];
for j = 1 : 40
    R1 = Quaternion2Rotation(x(4:7));
    C1 = x(1:3);

    df_dc = [];
    df_dR = [];
    for k = 1 : nPoints
        df_dc = [df_dc; JacobianC(K, R1, C1, X(k,:)')];
        df_dR = [df_dR; JacobianR(K, R1, C1, X(k,:)')*JacobianQ(x(4:7))];
    end

    u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
    u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];

    jacobian = [df_dc df_dR];
    delta_b = u(:)-u1(:);

    delta_x = inv(jacobian'*jacobian+lambda*eye(size(jacobian'*jacobian,1))*jacobian'*delta_b
    x = x + delta_x;
    x(4:7) = x(4:7)/norm(x(4:7));
end
```

$$\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}}\begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\frac{\partial u}{\partial \mathbf{p}} - u\frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v\frac{\partial u}{\partial \mathbf{p}} - v\frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

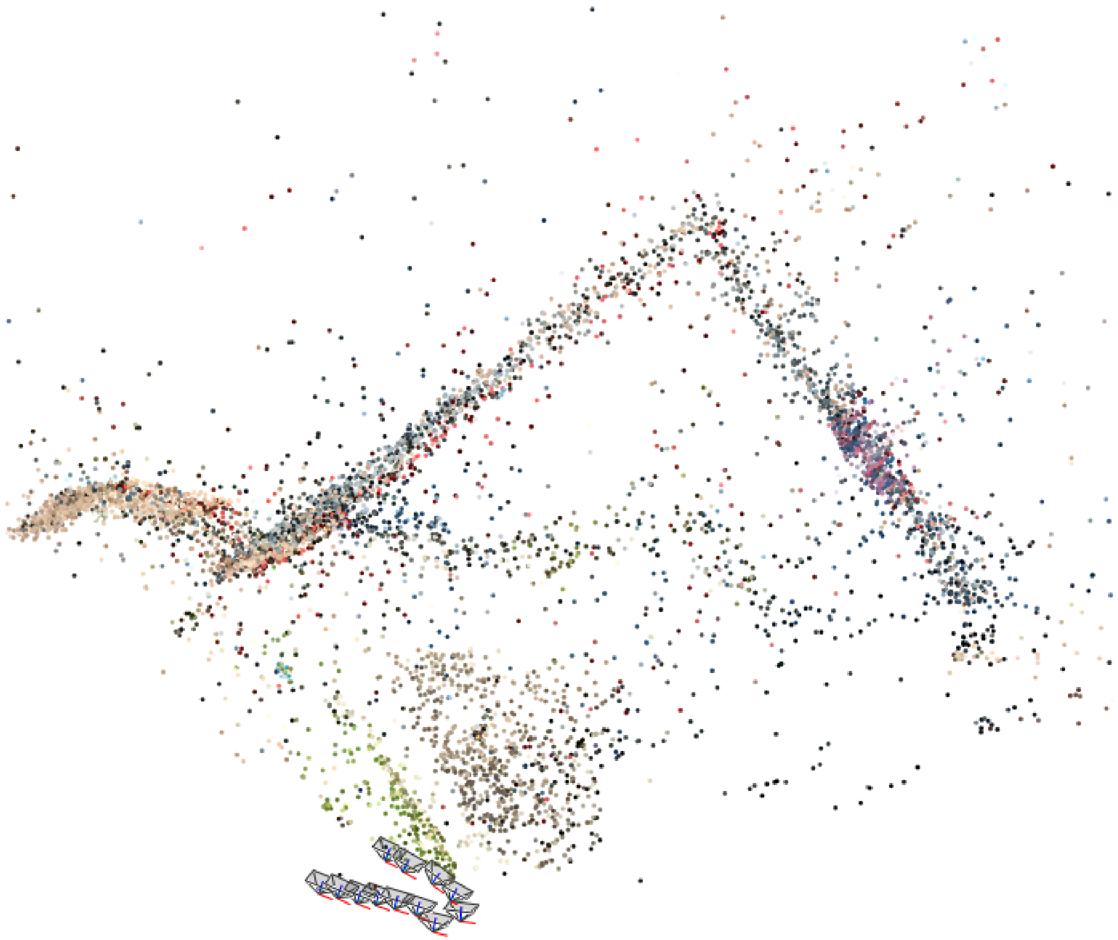
Example




```
u = K*R*[eye(3) -C]*[X'; ones(1,nPoints)];
u = [u(1,:)./u(3,:); u(2,:)./u(3,:)];

x = [C; q];
for j = 1 : 40
    R1 = Quaternion2Rotation(x(4:7));
    C1 = x(1:3);

    df_dc = [];
    df_dR = [];
    for k = 1 : nPoints
        df_dc = [df_dc; JacobianC(K, R1, C1, X(k,:)')];
        df_dR = [df_dR; JacobianR(K, R1, C1, X(k,:)')*JacobianQ(x(4:7))];
    end

    u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
    u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];

    jacobian = [df_dc df_dR];
    delta_b = u(:)-u1(:);

    delta_x = inv(jacobian'*jacobian+lambda*eye(size(jacobian'*jacobian,1)))*jacobian'*delta_b
    x = x + delta_x;
    x(4:7) = x(4:7)/norm(x(4:7));
end
```

$$\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}}\begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\frac{\partial u}{\partial \mathbf{p}} - u\frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v\frac{\partial u}{\partial \mathbf{p}} - v\frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

$$\Delta\mathbf{p} = \left(\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}^{\mathsf{T}}\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} + \lambda\mathbf{I}\right)^{-1}\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}^{\mathsf{T}}\left(\mathbf{b} - f(\mathbf{p})\right)$$



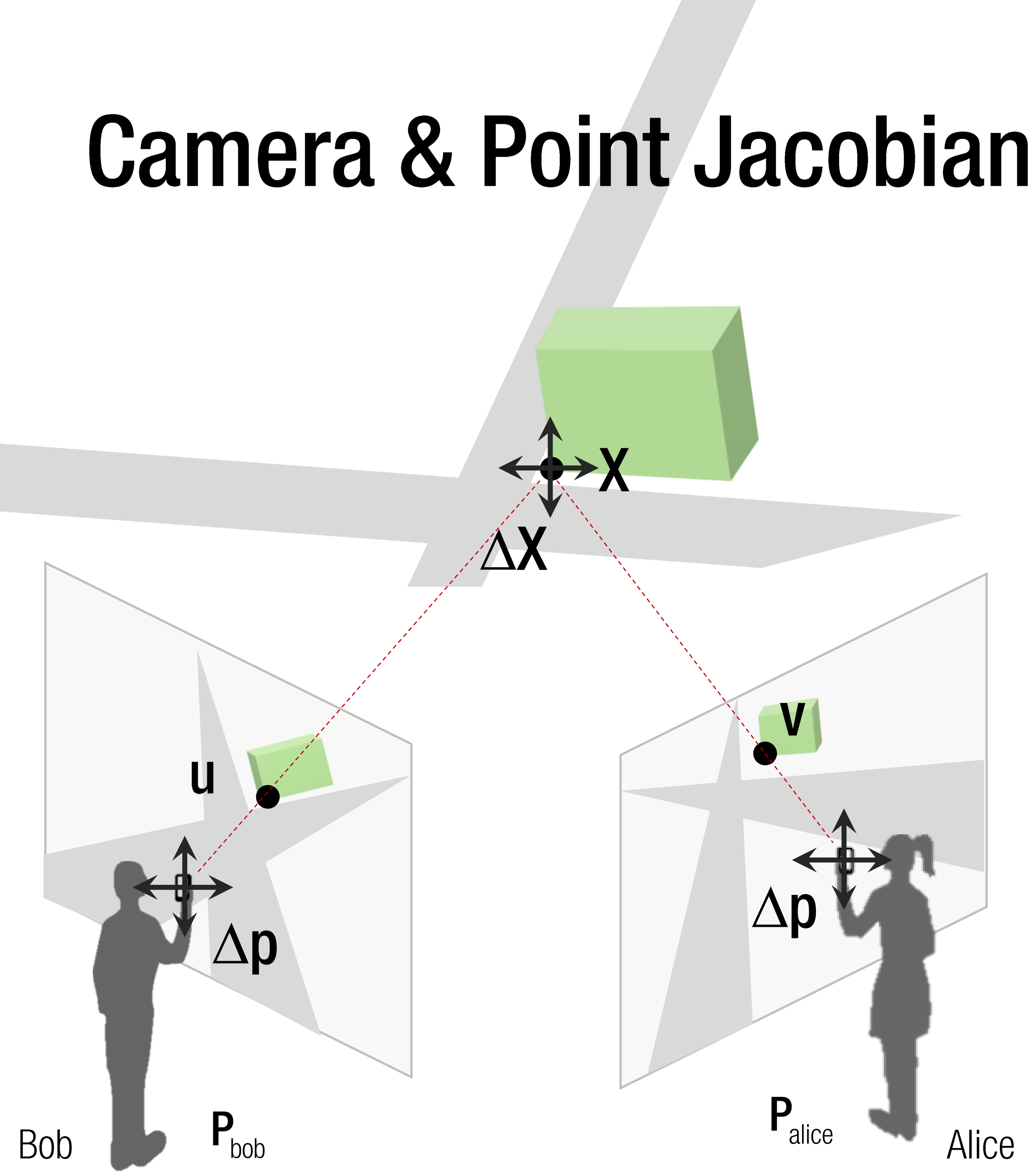
Bundle Adjustment

Black: given variables
Red: unknowns

Camera & Point Jacobian

$$E_{geom} = \left\|\hat{\mathbf{u}} - \mathbf{u}\right\|^2$$

$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - x\right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - y\right)^2$$

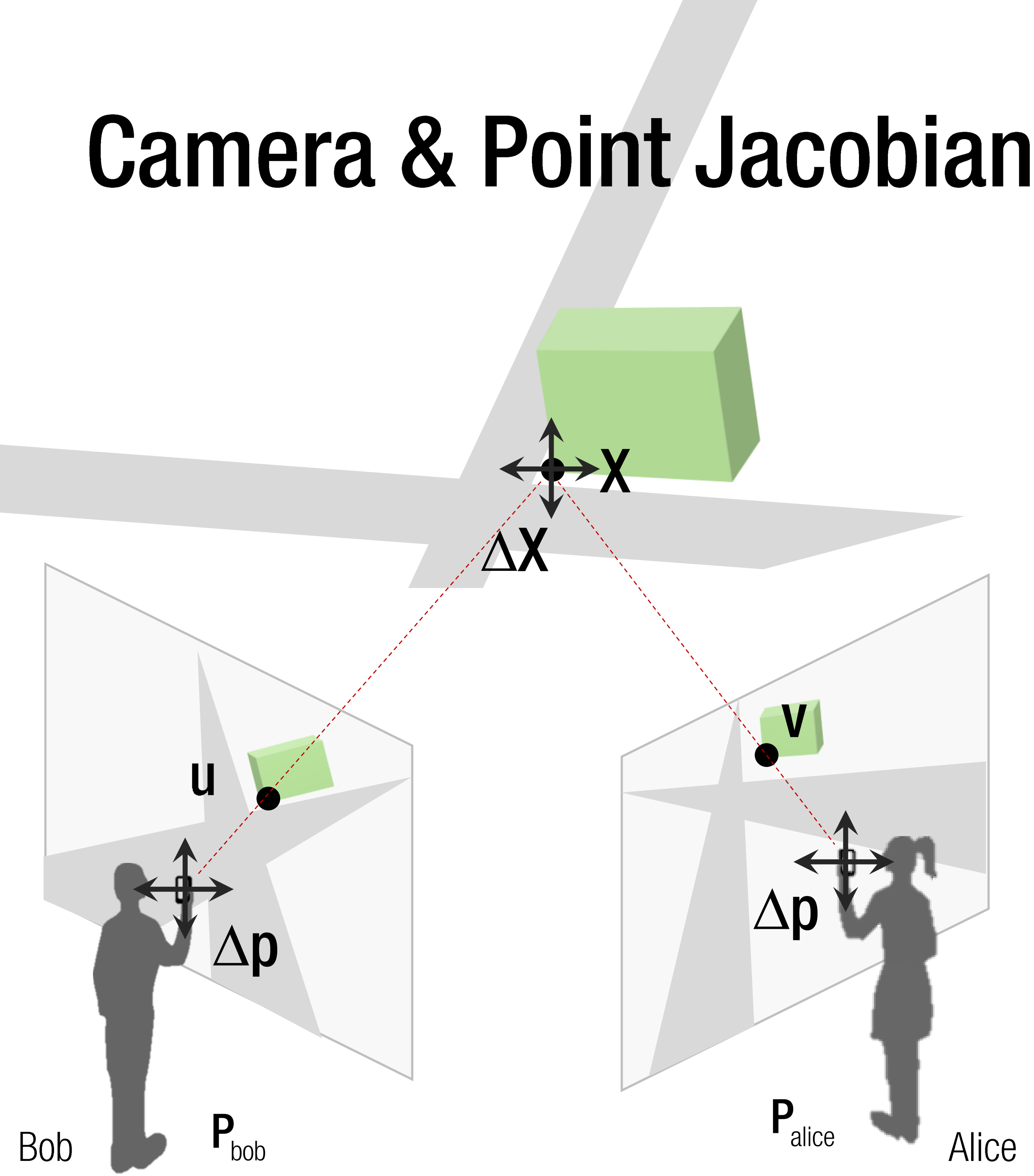


Black: given variables
Red: unknowns

Camera & Point Jacobian

$$E_{geom} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

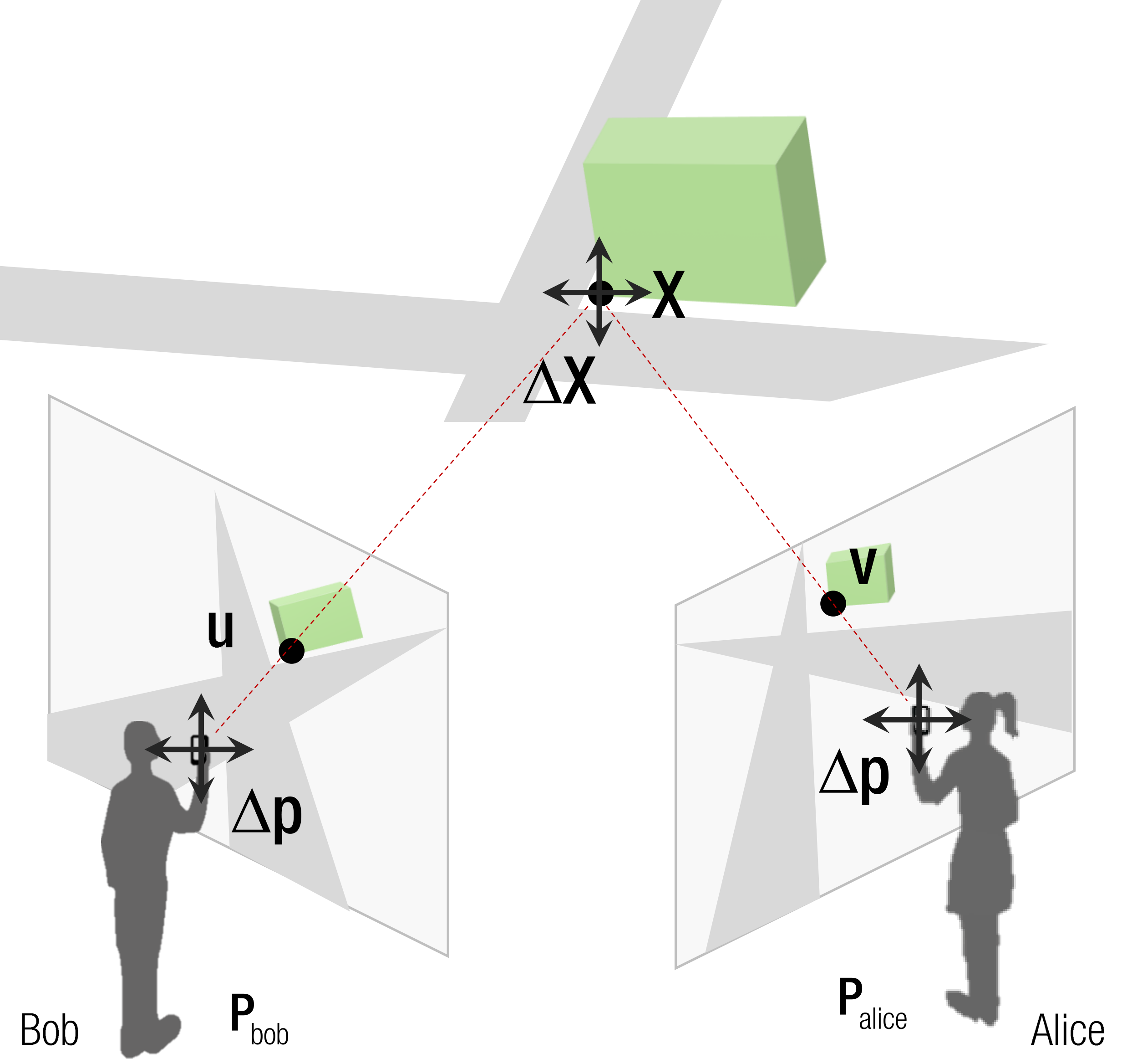
$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - y \right)^2$$



$$f(\mathbf{p},\mathbf{X}) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix}$$

Camera & Point Jacobian

Black: given variables
Red: unknowns



$$E_{\mathrm{geom}} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - y \right)^2$$

$$f(\mathbf{p},\mathbf{X}) = \begin{bmatrix} \dfrac{u}{w} \\ \dfrac{v}{w} \end{bmatrix} \longrightarrow \frac{\partial f(\mathbf{p},\mathbf{X})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} \dfrac{u}{w} \\ \dfrac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\dfrac{\partial u}{\partial \mathbf{p}} - u\dfrac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v\dfrac{\partial u}{\partial \mathbf{p}} - v\dfrac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

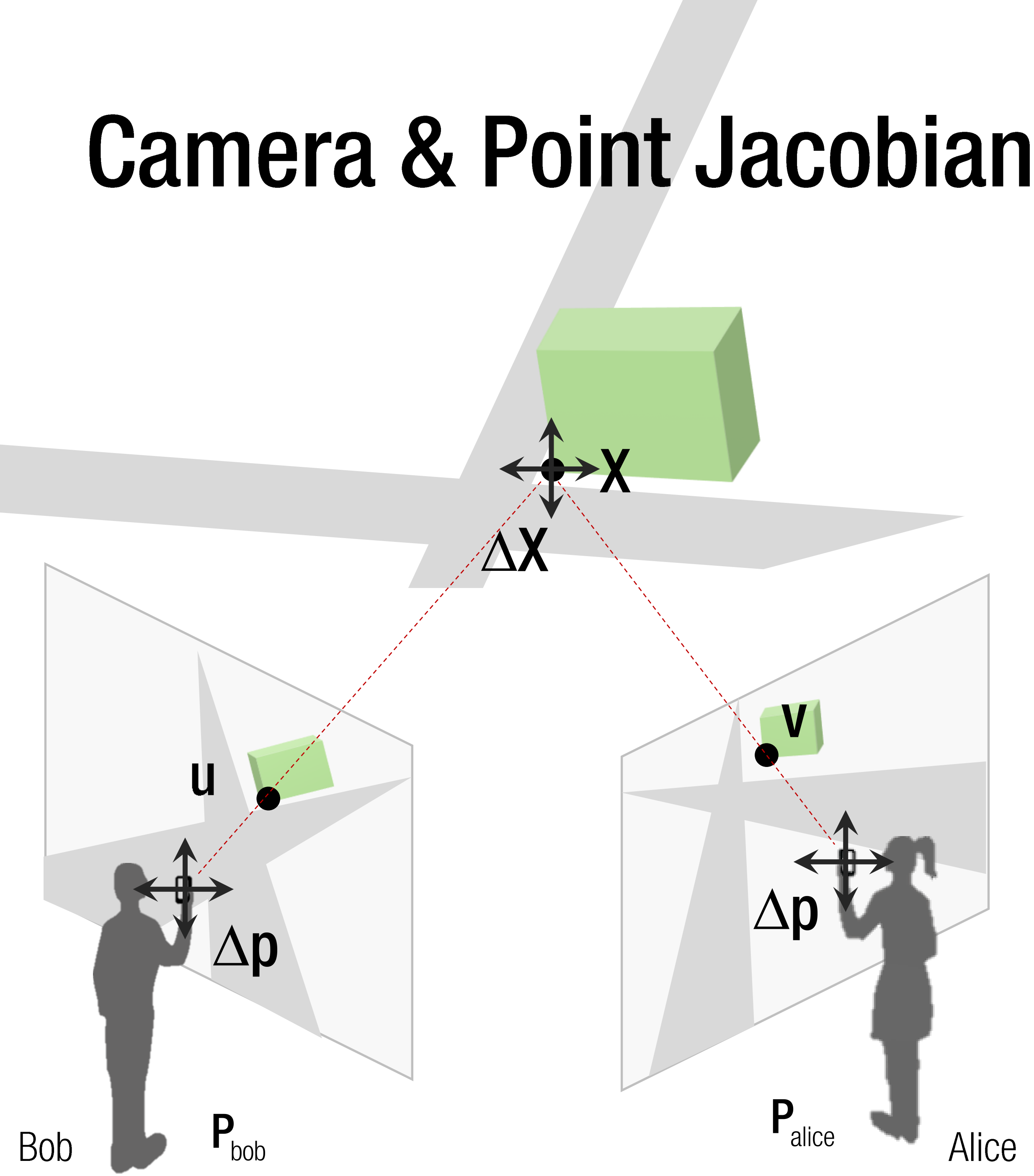
$$\longrightarrow \frac{\partial f(\mathbf{p},\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} \dfrac{u}{w} \\ \dfrac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w\dfrac{\partial u}{\partial \mathbf{X}} - u\dfrac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \dfrac{v\dfrac{\partial u}{\partial \mathbf{X}} - v\dfrac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix}$$

Camera & Point Jacobian

Black: given variables
Red: unknowns

$$E_{\text{geom}} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

$$= \left(\frac{\mathbf{P}_1\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2\mathbf{X}}{\mathbf{P}_3\mathbf{X}} - y \right)^2$$

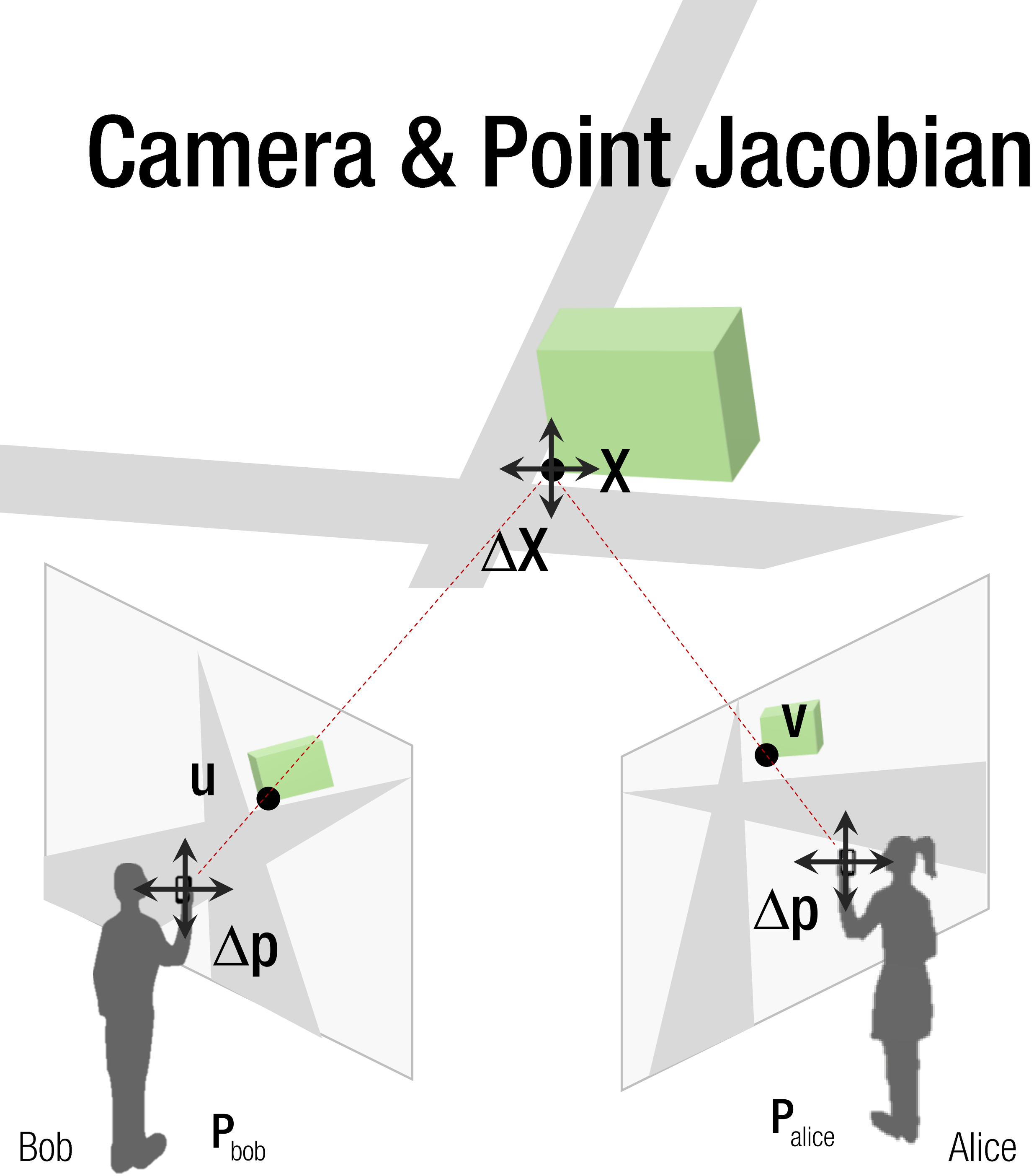


$$f(\mathbf{p}, \mathbf{X}) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} \longrightarrow \frac{\partial f(\mathbf{p}, \mathbf{X})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w \frac{\partial u}{\partial \mathbf{p}} - u \frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v \frac{\partial u}{\partial \mathbf{p}} - v \frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

Camera & Point Jacobian

Black: given variables

Red: unknowns



$$E_{\text{geom}} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

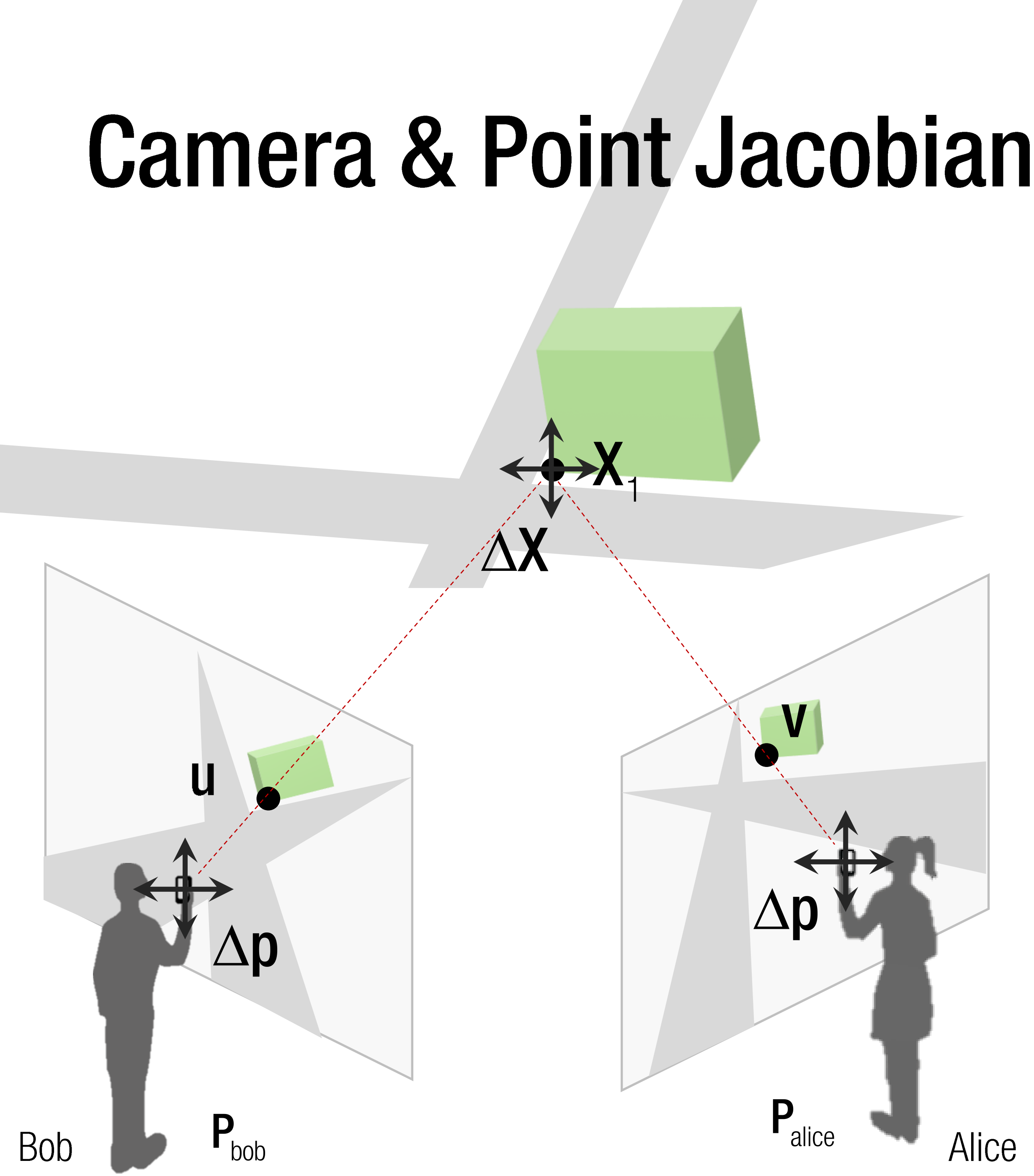
$$= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - y \right)^2$$

$$f(\mathbf{p}, \mathbf{X}) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} \longrightarrow \frac{\partial f(\mathbf{p}, \mathbf{X})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w \frac{\partial u}{\partial \mathbf{p}} - u \frac{\partial w}{\partial \mathbf{p}}}{w^2} \\ \dfrac{v \frac{\partial u}{\partial \mathbf{p}} - v \frac{\partial w}{\partial \mathbf{p}}}{w^2} \end{bmatrix}$$

$$\longrightarrow \frac{\partial f(\mathbf{p}, \mathbf{X})}{\partial \mathbf{X}} = \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} \dfrac{w \frac{\partial u}{\partial \mathbf{X}} - u \frac{\partial w}{\partial \mathbf{X}}}{w^2} \\ \dfrac{v \frac{\partial u}{\partial \mathbf{X}} - v \frac{\partial w}{\partial \mathbf{X}}}{w^2} \end{bmatrix}$$

Black: given variables
Red: unknowns

Camera & Point Jacobian

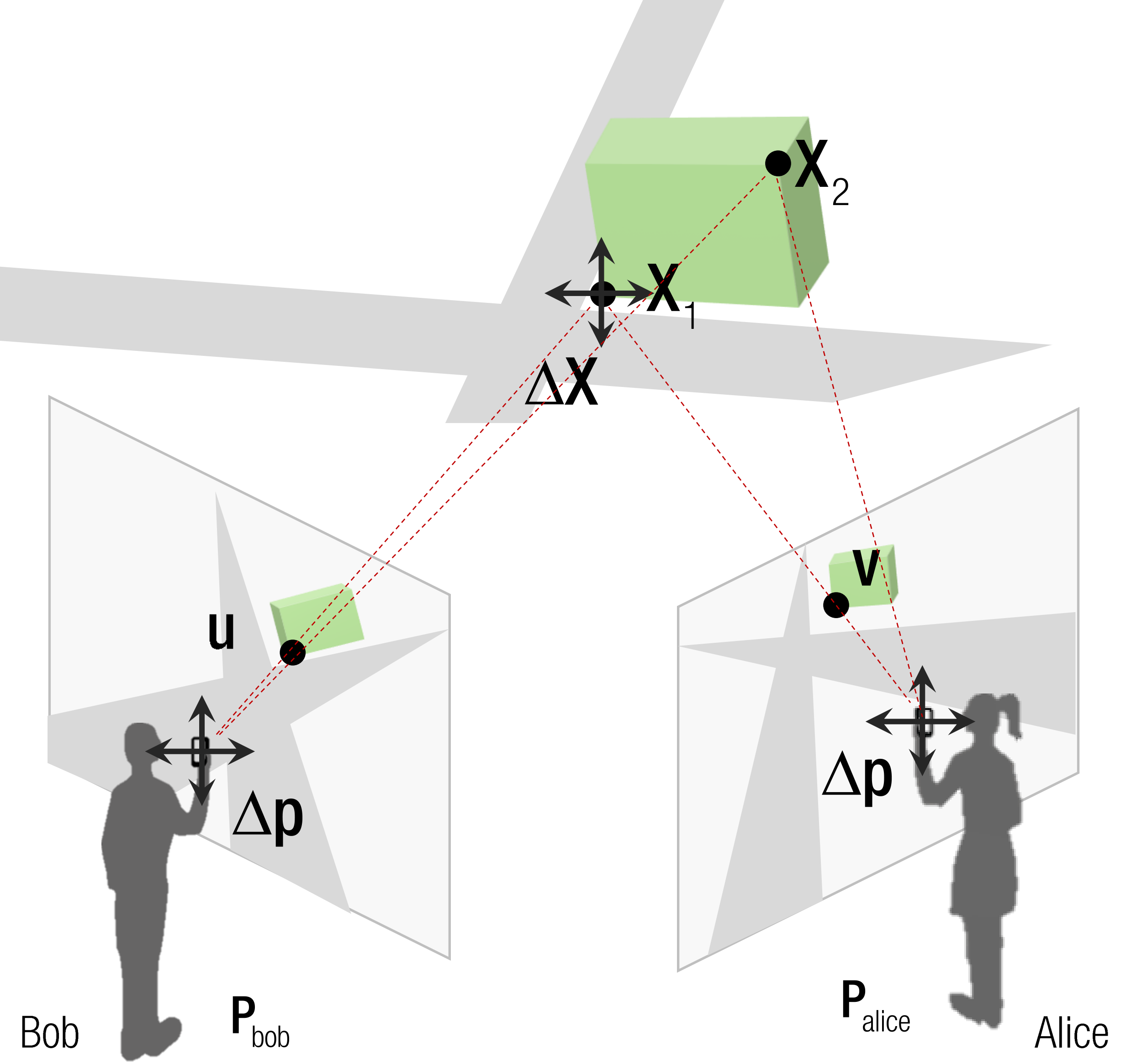


$$\mathbf{J}_{ij} = \begin{bmatrix} \dfrac{\partial f(\mathbf{p}_j, \mathbf{X}_i)}{\partial \mathbf{p}_j} & \dfrac{\partial f(\mathbf{p}_j, \mathbf{X}_i)}{\partial \mathbf{X}_i} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{\mathbf{p}ij} & \mathbf{J}_{\mathbf{X}ij} \end{bmatrix}$$

$$\mathbf{J} = \begin{matrix} \mathbf{J}_{\mathbf{p}1,bob} & \mathbf{0}_{2\times7} & \mathbf{J}_{\mathbf{X}1,bob} \\ \mathbf{0}_{2\times7} & \mathbf{J}_{\mathbf{p}1,alice} & \mathbf{J}_{\mathbf{X}1,alice} \end{matrix}$$

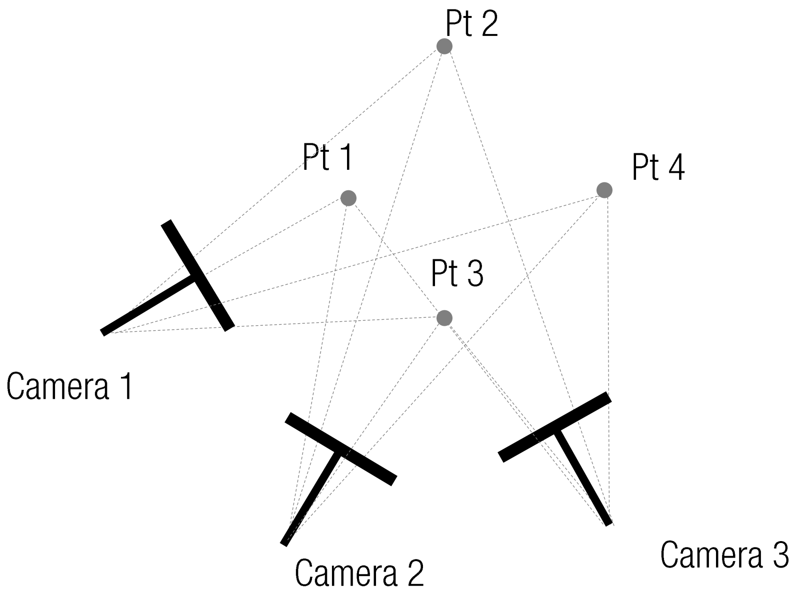
Camera & Point Jacobian

Black: given variables
Red: unknowns

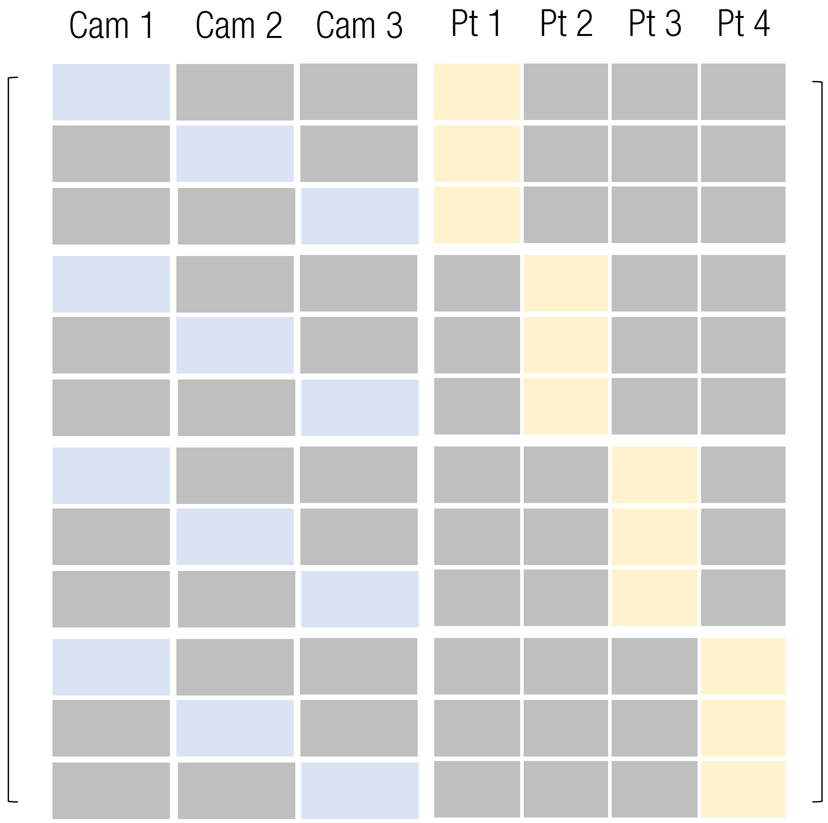


$$\mathbf{J}_{ij} = \begin{bmatrix} \dfrac{\partial f(\mathbf{p}_j, \mathbf{X}_i)}{\partial \mathbf{p}_j} & \dfrac{\partial f(\mathbf{p}_j, \mathbf{X}_i)}{\partial \mathbf{X}_i} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{\mathbf{p}ij} & \mathbf{J}_{\mathbf{X}ij} \end{bmatrix}$$

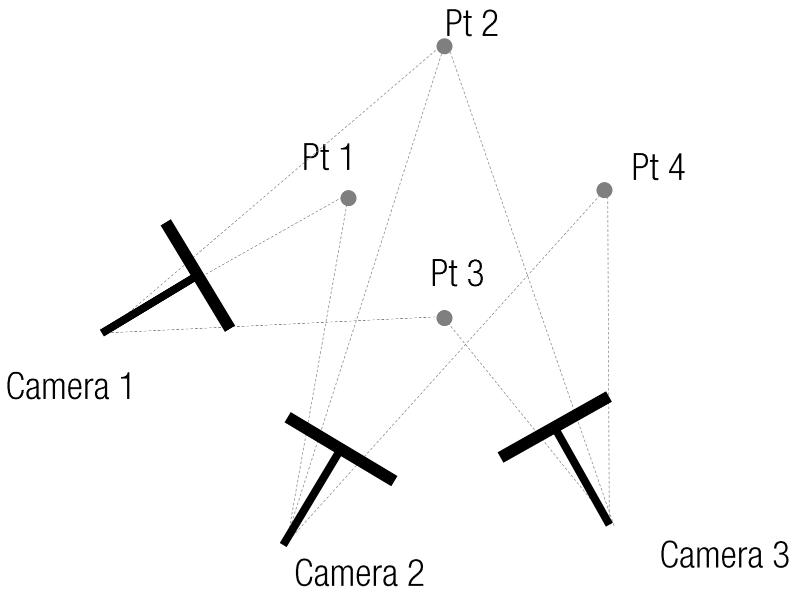
$$\mathbf{J} = \begin{matrix} \mathbf{J}_{\mathbf{p}1,bob} & \mathbf{0}_{2\times7} & \mathbf{J}_{\mathbf{X}1,bob} & \mathbf{0}_{2\times3} \\ \mathbf{0}_{2\times7} & \mathbf{J}_{\mathbf{p}1,alice} & \mathbf{J}_{\mathbf{X}1,alice} & \mathbf{0}_{2\times3} \\ \mathbf{J}_{\mathbf{p}2,bob} & \mathbf{0}_{2\times7} & \mathbf{0}_{2\times3} & \mathbf{J}_{\mathbf{X}2,bob} \\ \mathbf{0}_{2\times7} & \mathbf{J}_{\mathbf{p}2,alice} & \mathbf{0}_{2\times3} & \mathbf{J}_{\mathbf{X}2,alice} \end{matrix}$$

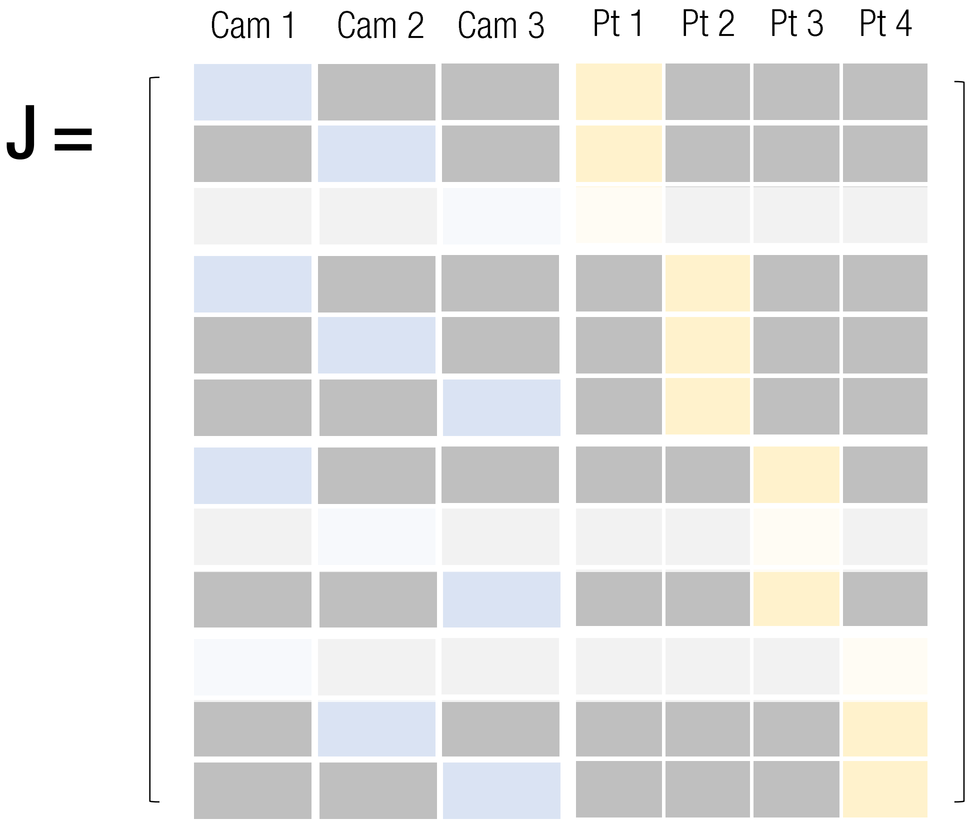


J =



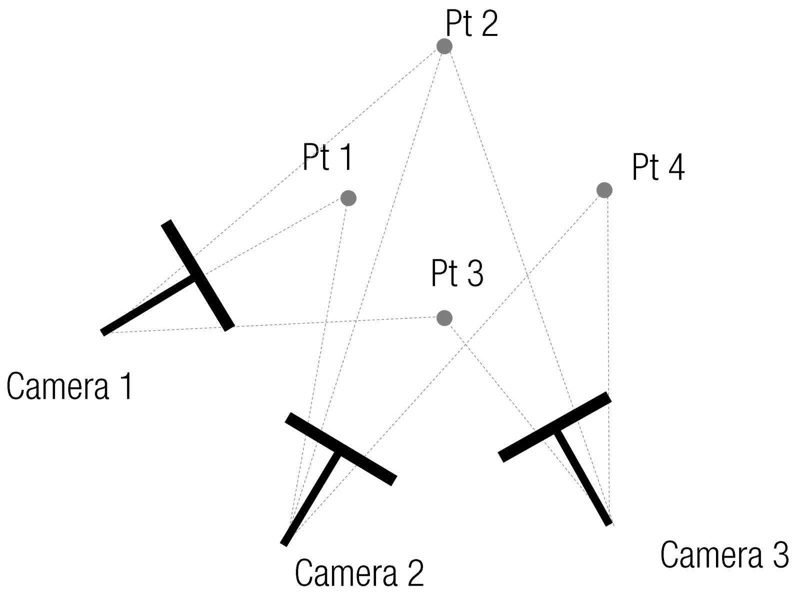
# of unknowns: 3x7+4x3
# of projections: 3x4

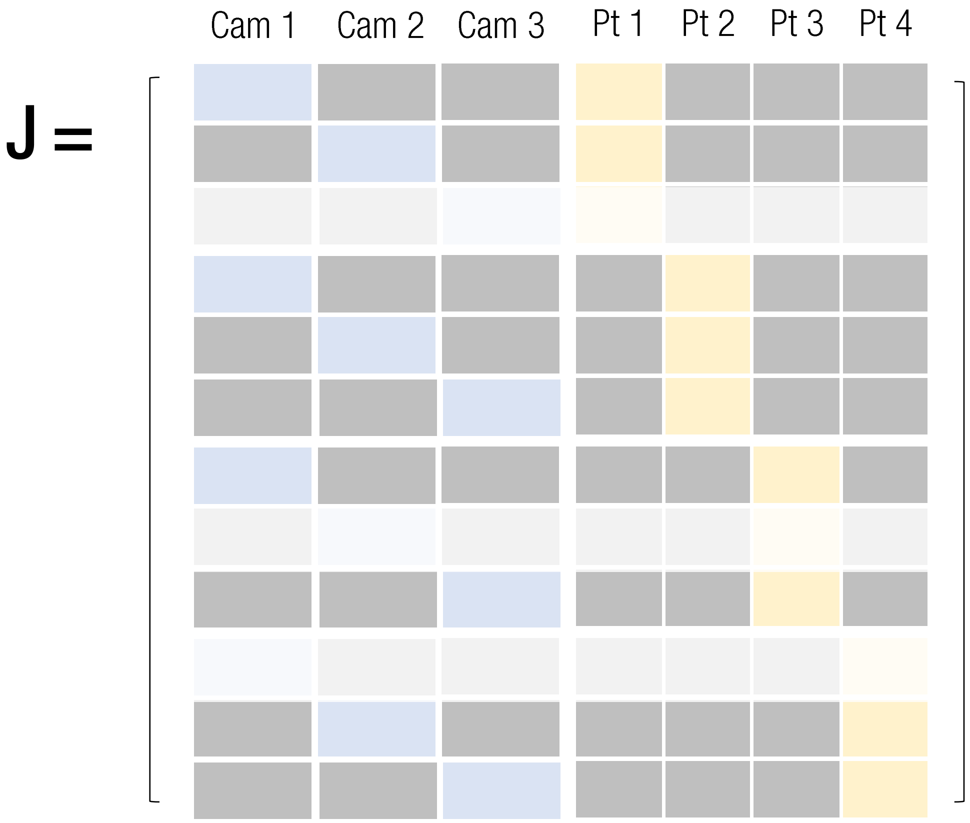




# of unknowns: 3x7+4x3

# of projections: 9 (not all points are visible from cameras)



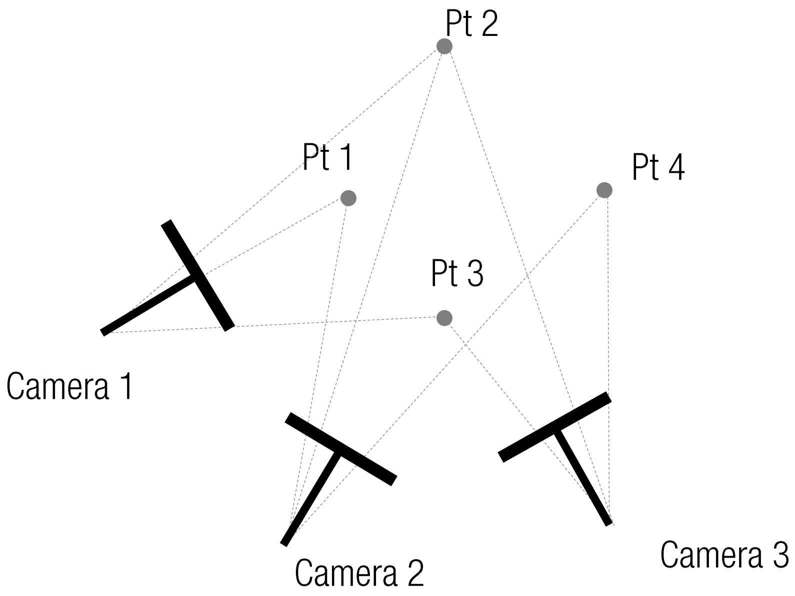


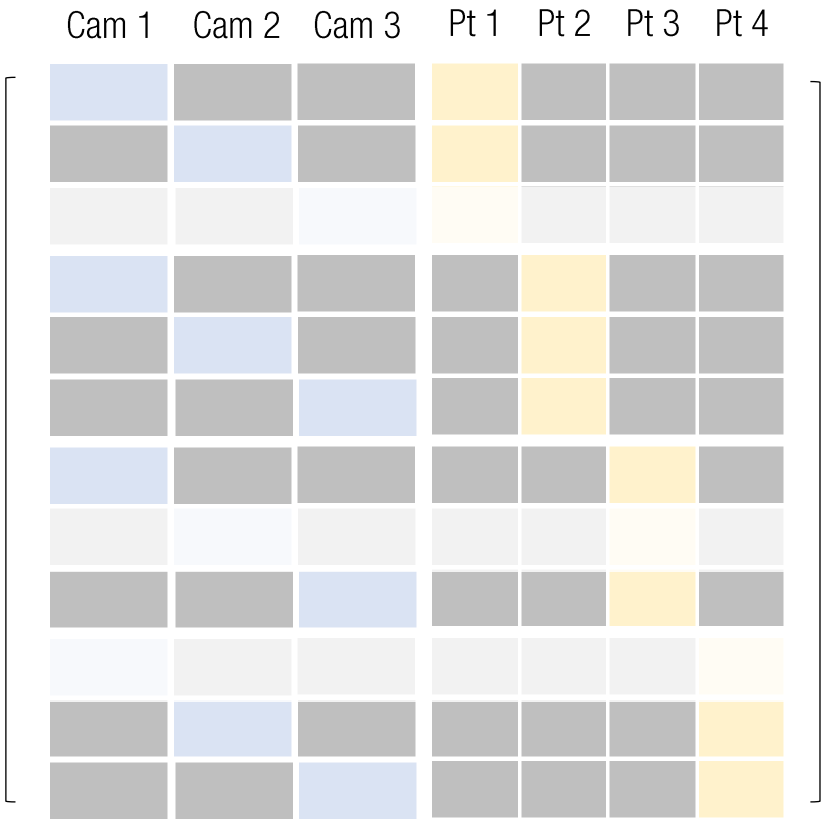
\# of unknowns: 3x7+4x3

\# of projections: 9 (not all points are visible from cameras)

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

size of $\mathbf{J}^{\mathrm{T}}\mathbf{J}$: 24x24



J = 

\# of unknowns: 3x7+4x3

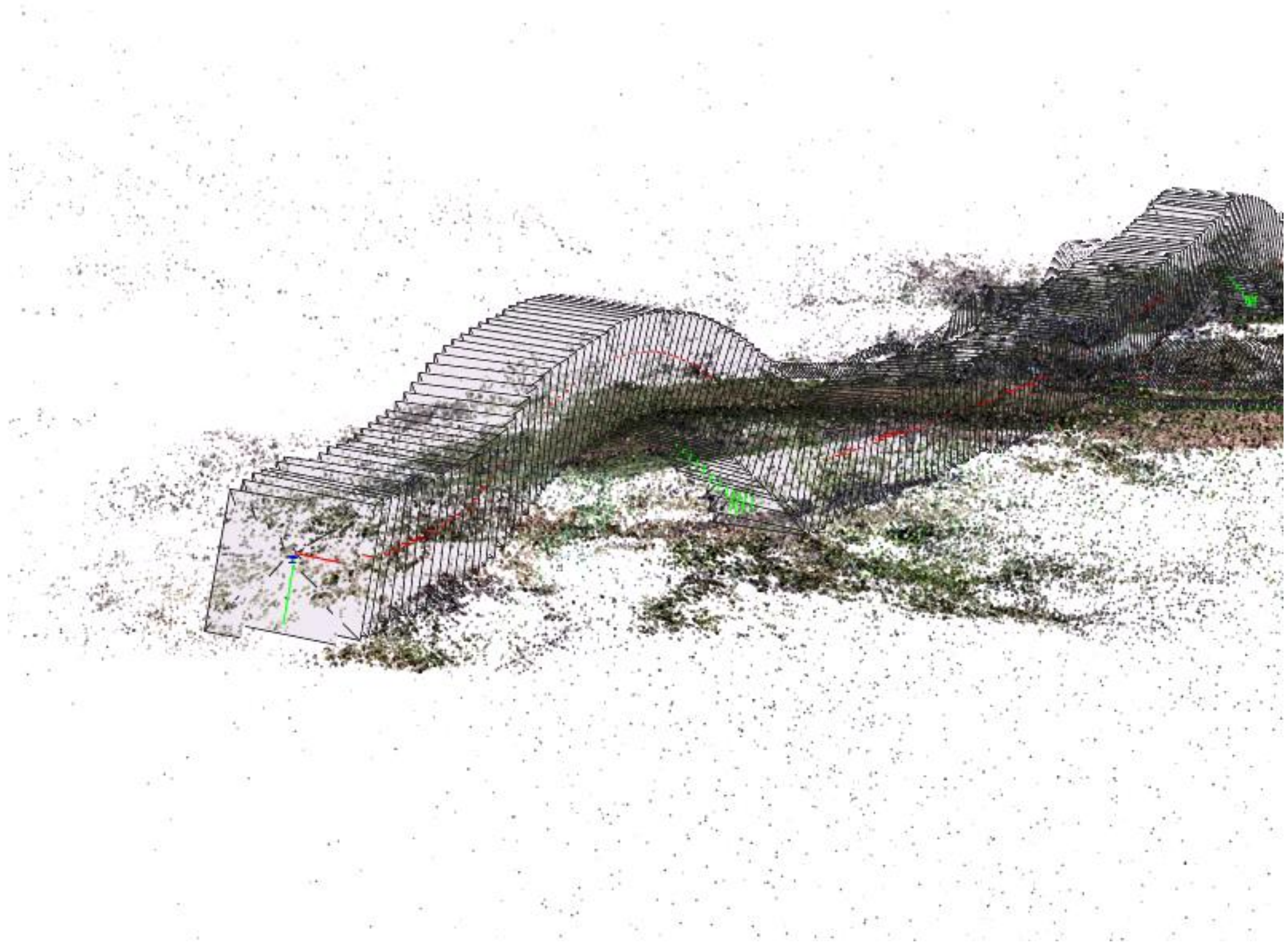
\# of projections: 9 (not all points are visible from cameras)

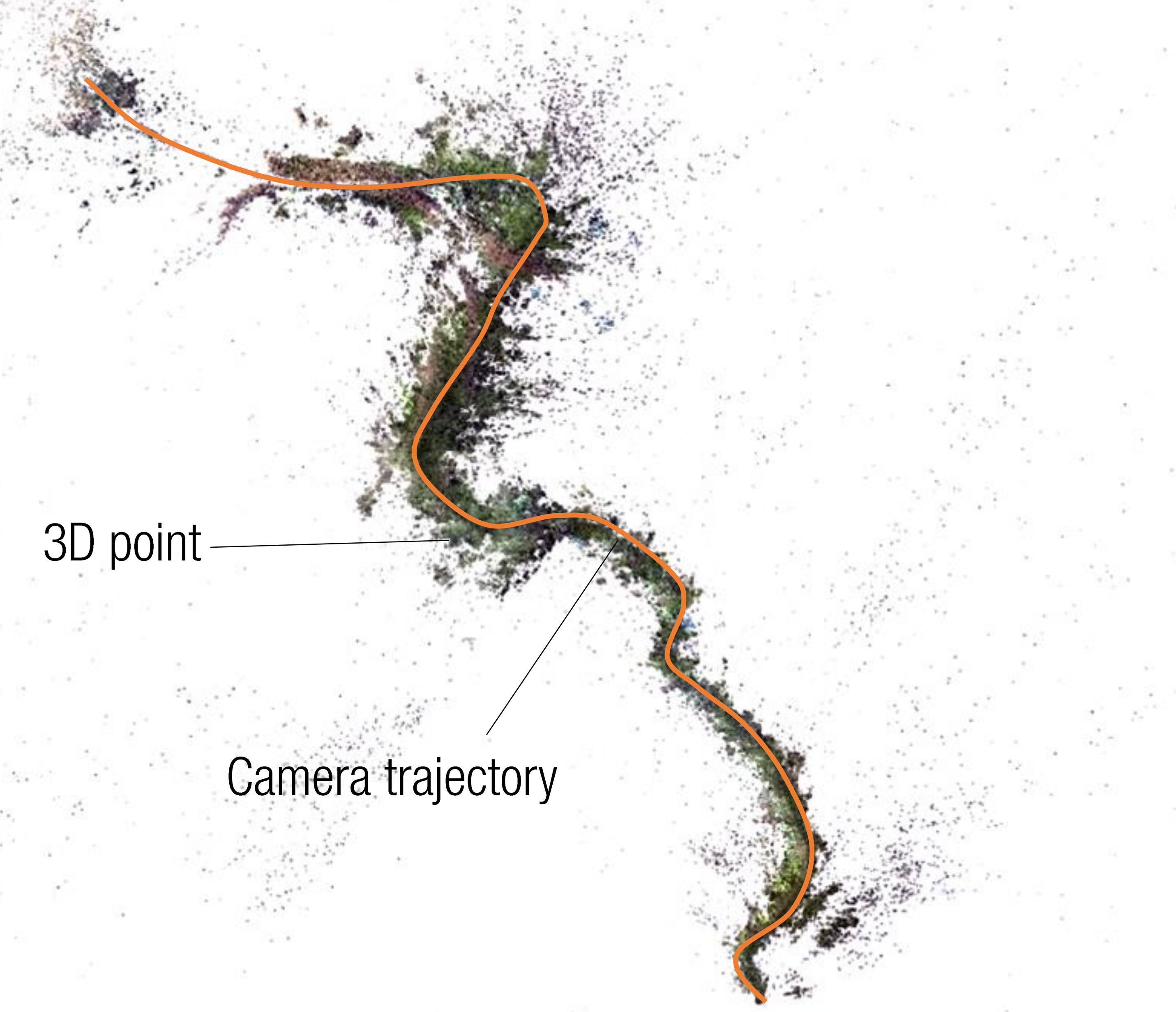
$$\Delta \mathbf{x} = \underline{\left(\mathbf{J}^{\mathrm{T}}\mathbf{J}\right)^{-1}} \mathbf{J}^{\mathrm{T}} \left(\mathbf{b} - \mathbf{f}(\mathbf{x})\right)$$

Main computational bottle neck

size of $\mathbf{J}^{\mathrm{T}}\mathbf{J}$: 24x24

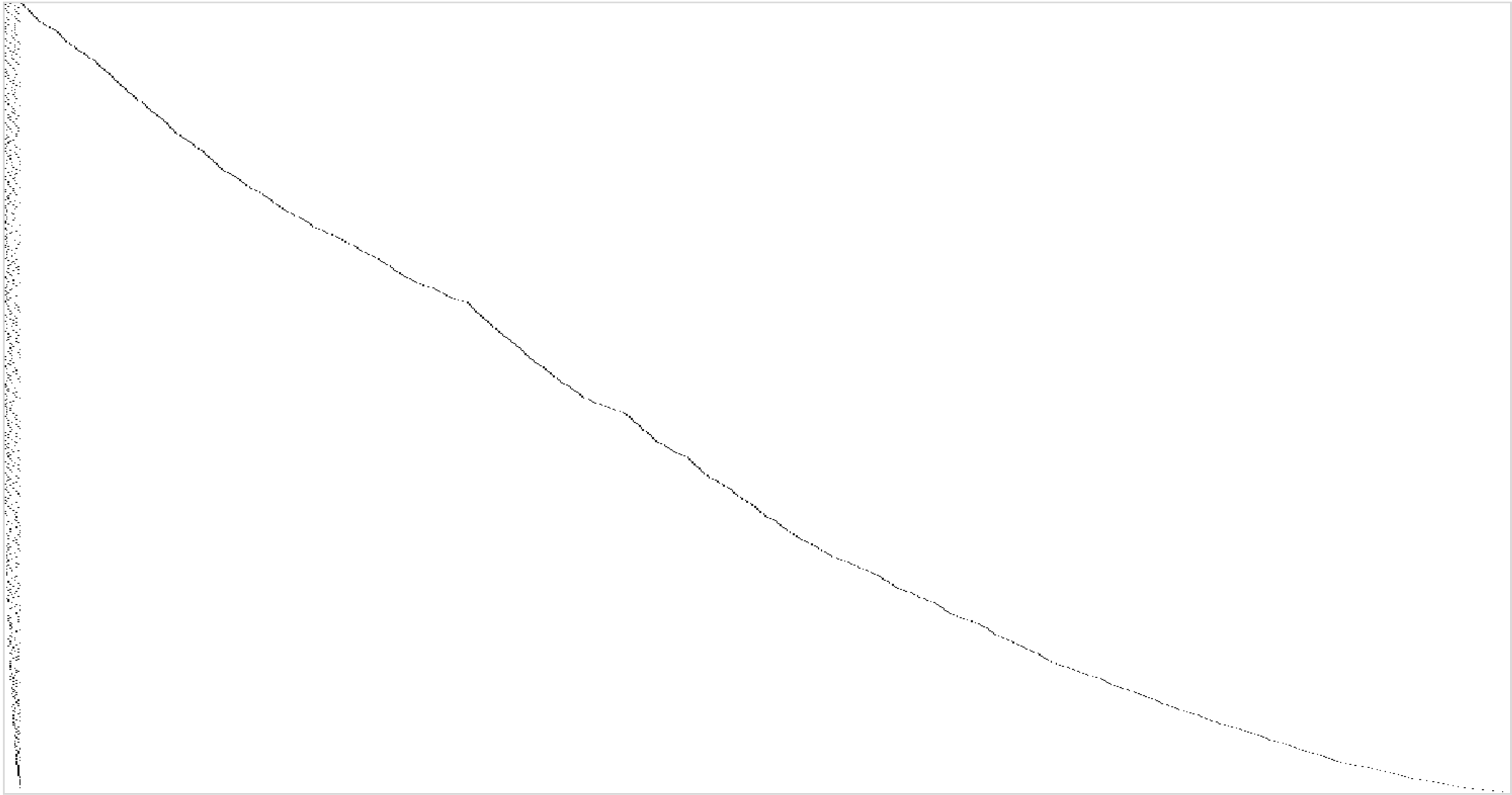

Input: first person video
https://www.youtube.com/watch?v=bKU0Z4bj7Mw

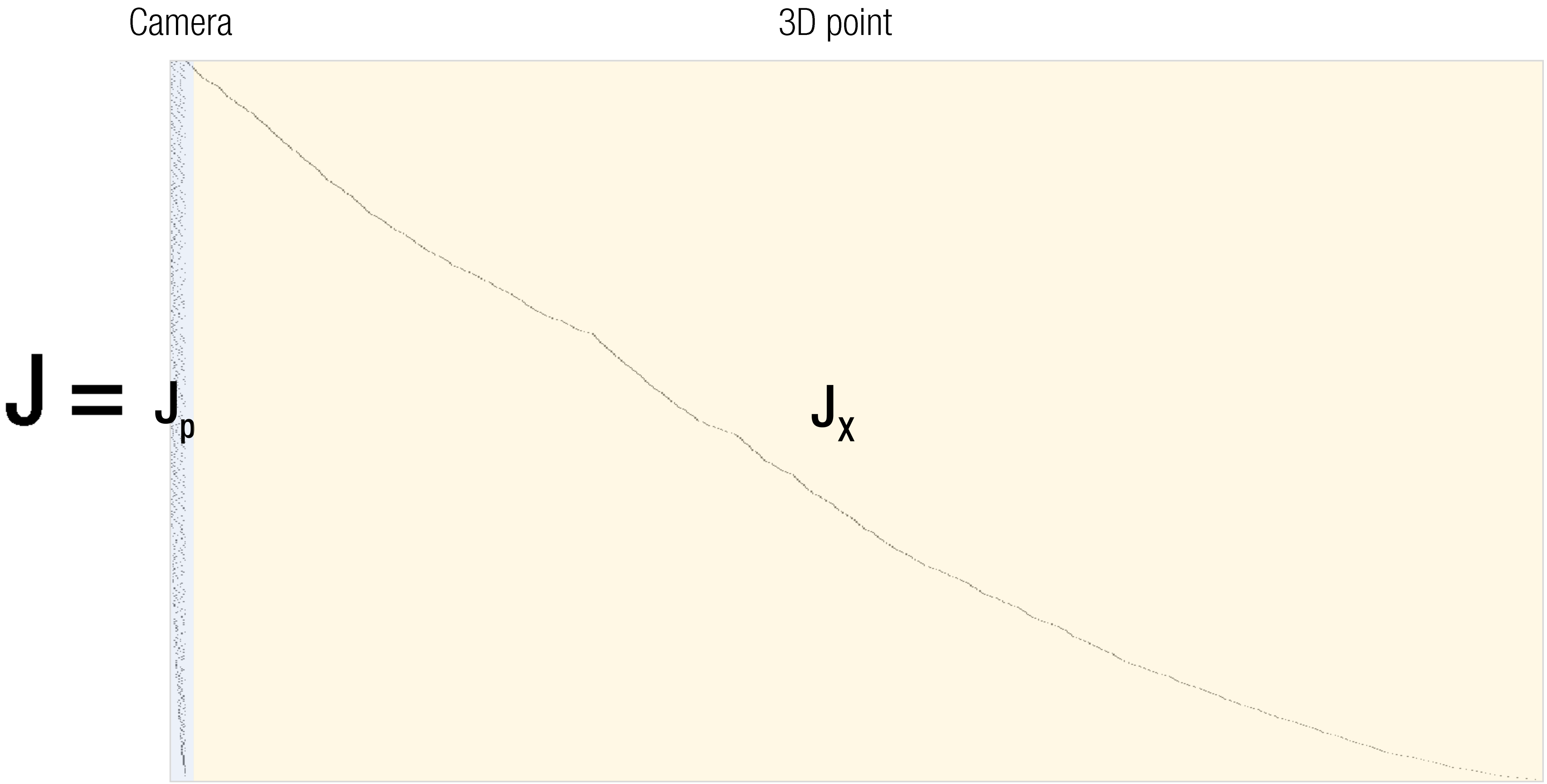




\# of cameras: 3009
\# of 3D points: 1,298,317

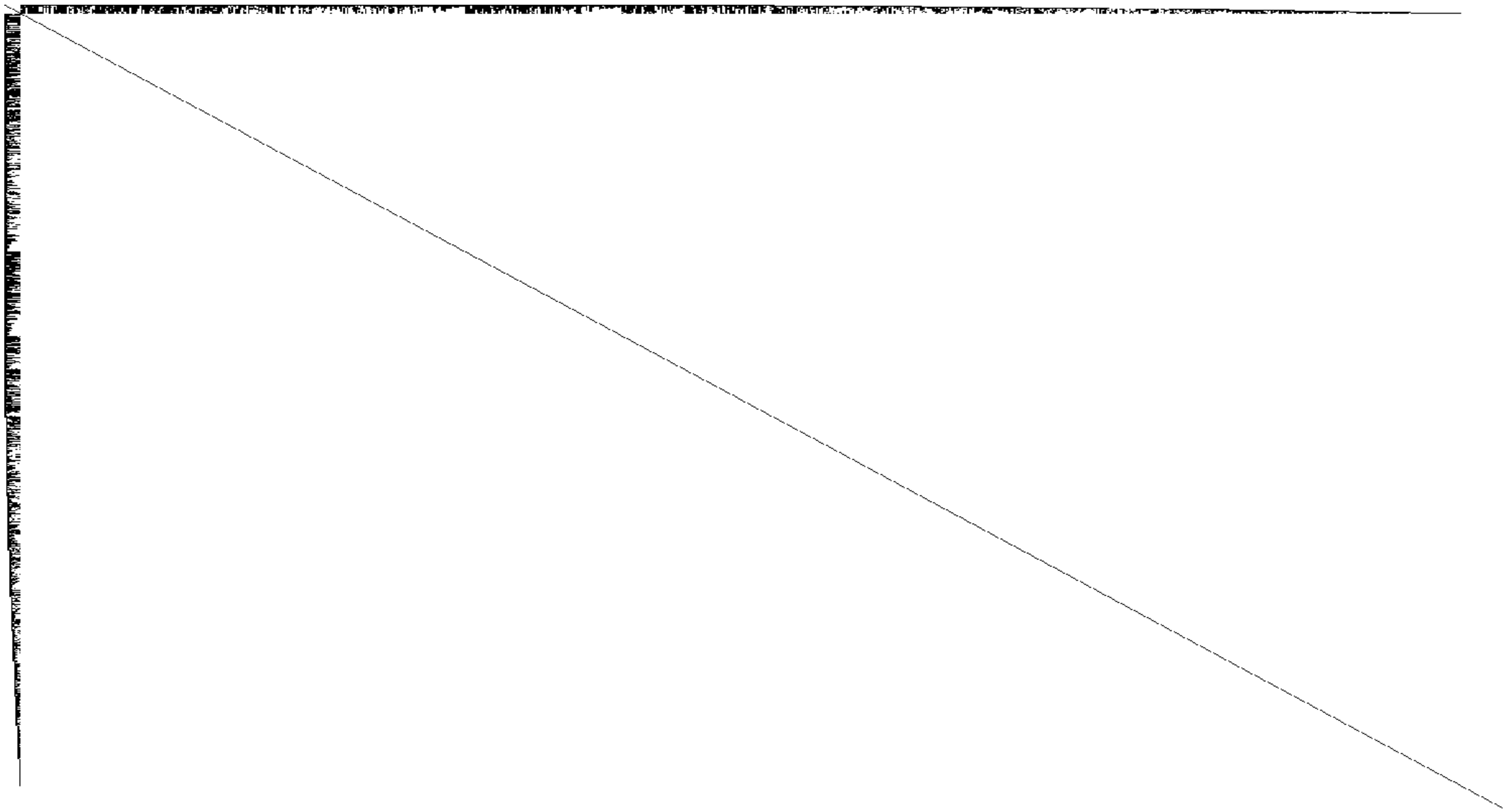
\# of unknowns: 7x3009+3x1,298,317
size of $\mathbf{J}^{\top}\mathbf{J}$: 6,167,690x6,167,690

$J =$ 



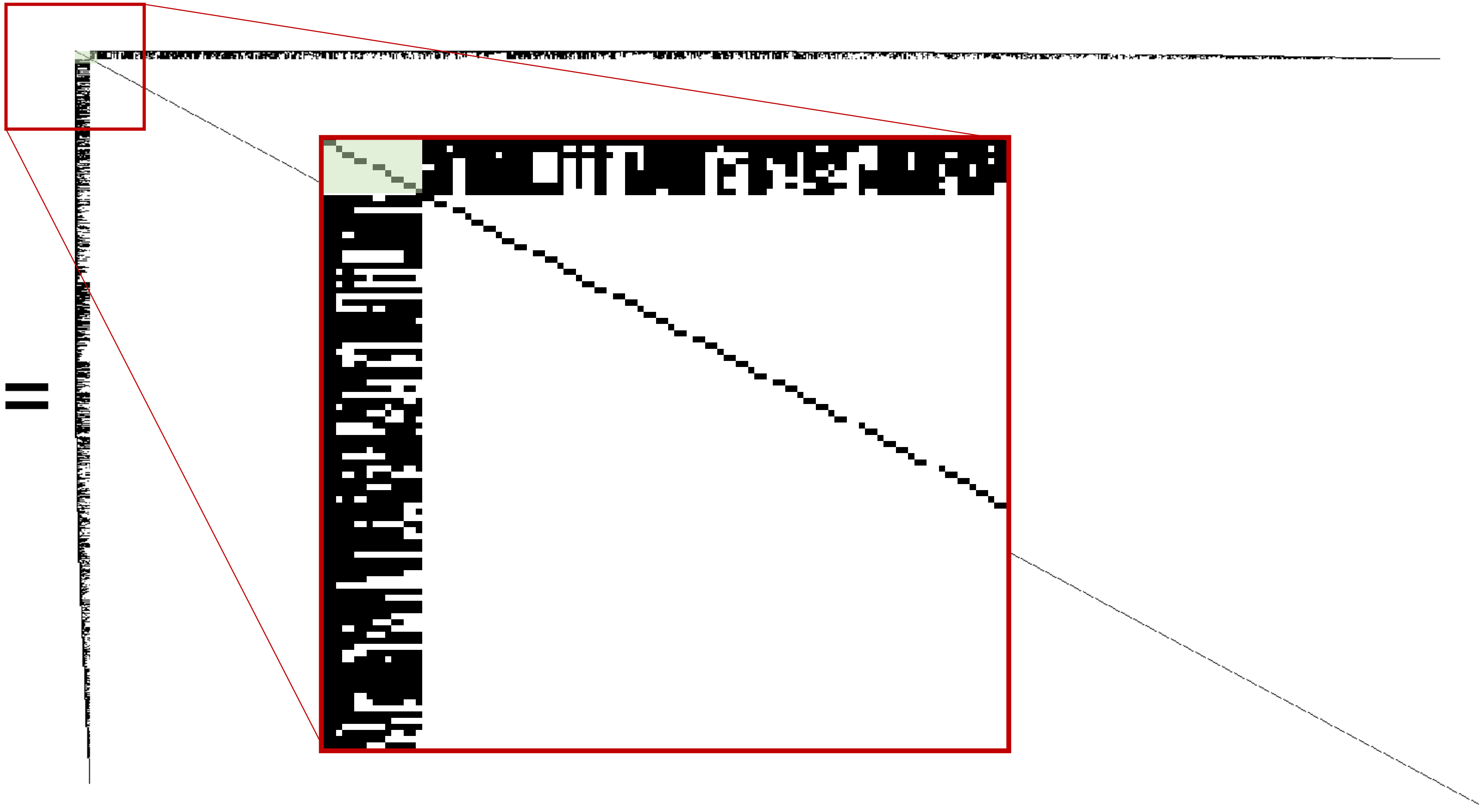
$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_p & \mathbf{J}_X \end{bmatrix}$$

$$\mathbf{J}^\top \mathbf{J} =$$



$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_p & \mathbf{J}_X \end{bmatrix} \qquad \mathbf{J}^\top \mathbf{J} = \begin{bmatrix} \mathbf{J}_p^\top \mathbf{J}_p & \mathbf{J}_p^\top \mathbf{J}_X \\ \mathbf{J}_X^\top \mathbf{J}_p & \mathbf{J}_X^\top \mathbf{J}_X \end{bmatrix}$$

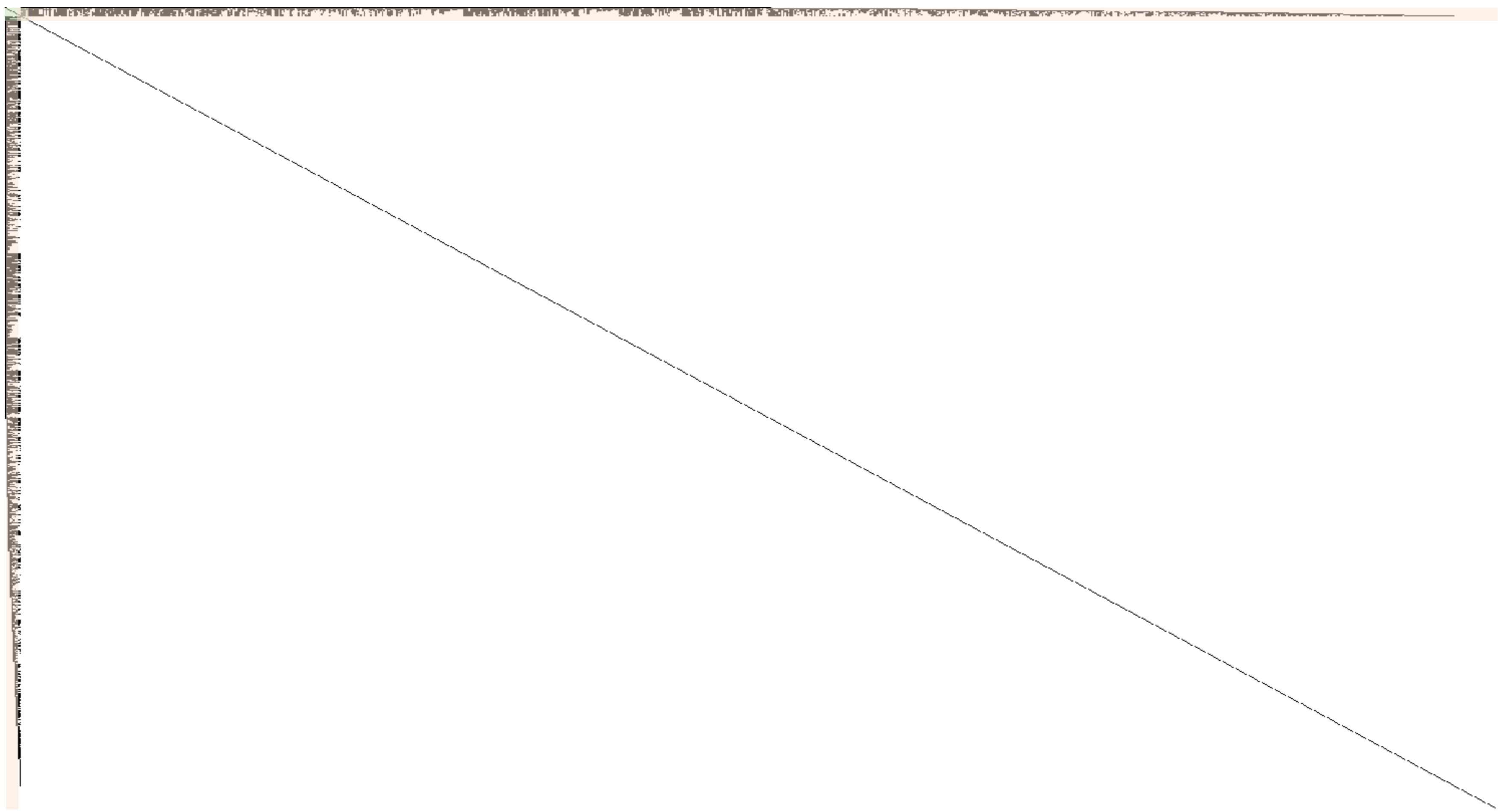
$$J^\top J =$$



$$J = \begin{bmatrix} J_p & J_x \end{bmatrix}$$

$$J^\top J = \begin{bmatrix} J_p^\top J_p & J_p^\top J_x \\ J_x^\top J_p & J_x^\top J_x \end{bmatrix} = \begin{bmatrix} A & B \\ B^\top & D \end{bmatrix}$$

$$J^\top J =$$



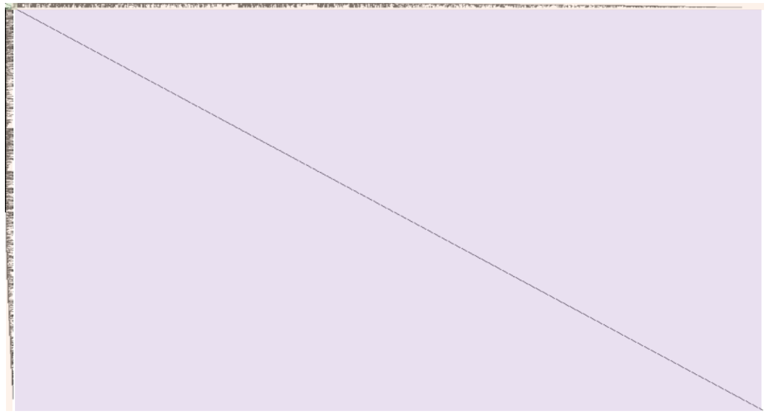
$$J = \begin{bmatrix} J_p & J_X \end{bmatrix}$$

$$J^\top J = \begin{bmatrix} J_p^\top J_p & J_p^\top J_X \\ J_X^\top J_p & J_X^\top J_X \end{bmatrix} = \begin{bmatrix} A & B \\ B^\top & D \end{bmatrix}$$

Normal equation:

$$\mathbf{J}^{\top}\mathbf{J}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix}=\mathbf{J}^{\top}\left(\mathbf{b}-f(\mathbf{X})\right)$$

$\mathbf{J}^{\top}\mathbf{J}=$

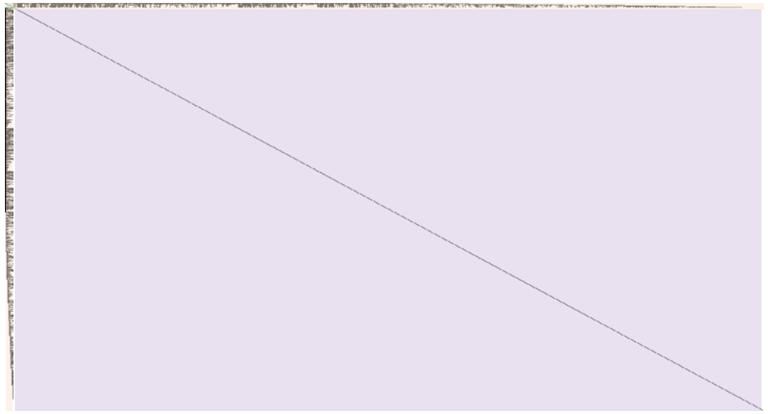


$$\mathbf{J}^{\top}\mathbf{J}=\begin{bmatrix}\mathbf{J}_{\mathbf{p}}^{\top}\mathbf{J}_{\mathbf{p}} & \mathbf{J}_{\mathbf{p}}^{\top}\mathbf{J}_{\mathbf{X}}\\ \mathbf{J}_{\mathbf{X}}^{\top}\mathbf{J}_{\mathbf{p}} & \mathbf{J}_{\mathbf{X}}^{\top}\mathbf{J}_{\mathbf{X}}\end{bmatrix}=\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\top} & \mathbf{D}\end{bmatrix}$$

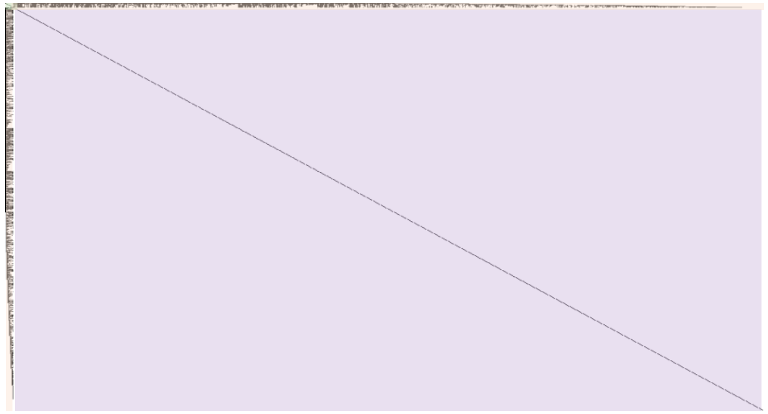
Normal equation:

$$\mathbf{J}^{\mathsf{T}}\mathbf{J}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \mathbf{J}^{\mathsf{T}}\left(\mathbf{b} - f(\mathbf{X})\right)$$

$$\longrightarrow \begin{bmatrix}\mathbf{J}_p^{\mathsf{T}}\mathbf{J}_p & \mathbf{J}_p^{\mathsf{T}}\mathbf{J}_X\\ \mathbf{J}_X^{\mathsf{T}}\mathbf{J}_p & \mathbf{J}_X^{\mathsf{T}}\mathbf{J}_X\end{bmatrix}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \begin{bmatrix}\mathbf{J}_p^{\mathsf{T}}\\ \mathbf{J}_X^{\mathsf{T}}\end{bmatrix}\left(\mathbf{b} - f(\mathbf{X})\right) = \begin{bmatrix}\mathbf{e}_p\\ \mathbf{e}_X\end{bmatrix}$$

$\mathbf{J}^{\mathsf{T}}\mathbf{J} =$ 

$$\mathbf{J}^{\mathsf{T}}\mathbf{J} = \begin{bmatrix}\mathbf{J}_p^{\mathsf{T}}\mathbf{J}_p & \mathbf{J}_p^{\mathsf{T}}\mathbf{J}_X\\ \mathbf{J}_X^{\mathsf{T}}\mathbf{J}_p & \mathbf{J}_X^{\mathsf{T}}\mathbf{J}_X\end{bmatrix} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\mathsf{T}} & \mathbf{D}\end{bmatrix}$$

$\mathbf{J}^T\mathbf{J} =$ 

$$\mathbf{J}^T\mathbf{J} = \begin{bmatrix} \mathbf{J}_p^T\mathbf{J}_p & \mathbf{J}_p^T\mathbf{J}_X \\ \mathbf{J}_X^T\mathbf{J}_p & \mathbf{J}_X^T\mathbf{J}_X \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{D} \end{bmatrix}$$

Normal equation:

$$\mathbf{J}^T\mathbf{J}\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{X} \end{bmatrix} = \mathbf{J}^T\left(\mathbf{b} - f(\mathbf{X})\right)$$

$$\longrightarrow \begin{bmatrix} \mathbf{J}_p^T\mathbf{J}_p & \mathbf{J}_p^T\mathbf{J}_X \\ \mathbf{J}_X^T\mathbf{J}_p & \mathbf{J}_X^T\mathbf{J}_X \end{bmatrix}\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_p^T \\ \mathbf{J}_X^T \end{bmatrix}\left(\mathbf{b} - f(\mathbf{X})\right) = \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

or
$$\begin{bmatrix} \mathbf{J}_p^T\mathbf{J}_p + \mu\mathbf{I} & \mathbf{J}_p^T\mathbf{J}_X \\ \mathbf{J}_X^T\mathbf{J}_p & \mathbf{J}_X^T\mathbf{J}_X + \mu\mathbf{I} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

or
$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{D} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

$$\mathbf{J}^{\mathsf T}\mathbf{J} =$$

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}_1 & & \\ & \ddots & \\ & & \mathbf{d}_M \end{bmatrix} \longrightarrow \mathbf{D}^{-1} = \begin{bmatrix} \mathbf{d}_1^{-1} & & \\ & \ddots & \\ & & \mathbf{d}_M^{-1} \end{bmatrix}$$

Inversion of block diagonal matrix can be efficiently computed.

$$\mathbf{J}^{\mathsf T}\mathbf{J} = \begin{bmatrix} \mathbf{J}_p^{\mathsf T}\mathbf{J}_p & \mathbf{J}_p^{\mathsf T}\mathbf{J}_X \\ \mathbf{J}_X^{\mathsf T}\mathbf{J}_p & \mathbf{J}_X^{\mathsf T}\mathbf{J}_X \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^{\mathsf T} & \mathbf{D} \end{bmatrix}$$

Normal equation:

$$\mathbf{J}^{\mathsf T}\mathbf{J}\begin{bmatrix} \Delta \mathbf{p} \\ \Delta \mathbf{X} \end{bmatrix} = \mathbf{J}^{\mathsf T}\left(\mathbf{b} - f(\mathbf{X})\right)$$

$$\longrightarrow \begin{bmatrix} \mathbf{J}_p^{\mathsf T}\mathbf{J}_p & \mathbf{J}_p^{\mathsf T}\mathbf{J}_X \\ \mathbf{J}_X^{\mathsf T}\mathbf{J}_p & \mathbf{J}_X^{\mathsf T}\mathbf{J}_X \end{bmatrix}\begin{bmatrix} \Delta \mathbf{p} \\ \Delta \mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_p^{\mathsf T} \\ \mathbf{J}_X^{\mathsf T} \end{bmatrix}\left(\mathbf{b} - f(\mathbf{X})\right) = \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

or

$$\begin{bmatrix} \mathbf{J}_p^{\mathsf T}\mathbf{J}_p + \mu\mathbf{I} & \mathbf{J}_p^{\mathsf T}\mathbf{J}_X \\ \mathbf{J}_X^{\mathsf T}\mathbf{J}_p & \mathbf{J}_X^{\mathsf T}\mathbf{J}_X + \mu\mathbf{I} \end{bmatrix}\begin{bmatrix} \Delta \mathbf{p} \\ \Delta \mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

or

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^{\mathsf T} & \mathbf{D} \end{bmatrix}\begin{bmatrix} \Delta \mathbf{p} \\ \Delta \mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{I} & -\mathbf{B}\mathbf{D}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^{\mathsf T} & \mathbf{D} \end{bmatrix}\begin{bmatrix} \Delta \mathbf{p} \\ \Delta \mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & -\mathbf{B}\mathbf{D}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

Normal equation:

$$\mathbf{J}^{\mathrm{T}}\mathbf{J}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \mathbf{J}^{\mathrm{T}}\left(\mathbf{b} - f(\mathbf{X})\right)$$

$$\longrightarrow \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\mathrm{T}} & \mathbf{D}\end{bmatrix}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \begin{bmatrix}\mathbf{e}_{\mathrm{p}}\\ \mathbf{e}_{\mathrm{X}}\end{bmatrix}$$

$$\mathbf{J}^{\mathrm{T}}\mathbf{J} = \quad \mathbf{D} = \begin{bmatrix}\mathbf{d}_1 & & \\ & \ddots & \\ & & \mathbf{d}_M\end{bmatrix} \longrightarrow \mathbf{D}^{-1} = \begin{bmatrix}\mathbf{d}_1^{-1} & & \\ & \ddots & \\ & & \mathbf{d}_M^{-1}\end{bmatrix}$$

Inversion of block diagonal matrix can be efficiently computed.

$$\mathbf{J}^{\mathrm{T}}\mathbf{J} = \begin{bmatrix}\mathbf{J}_{\mathrm{p}}^{\mathrm{T}}\mathbf{J}_{\mathrm{p}} & \mathbf{J}_{\mathrm{p}}^{\mathrm{T}}\mathbf{J}_{\mathrm{X}}\\ \mathbf{J}_{\mathrm{X}}^{\mathrm{T}}\mathbf{J}_{\mathrm{p}} & \mathbf{J}_{\mathrm{X}}^{\mathrm{T}}\mathbf{J}_{\mathrm{X}}\end{bmatrix} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\mathrm{T}} & \mathbf{D}\end{bmatrix}$$

Normal equation:

$$\mathbf{J}^{\mathsf{T}}\mathbf{J}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \mathbf{J}^{\mathsf{T}}\left(\mathbf{b} - f(\mathbf{X})\right)$$

$$\longrightarrow \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\mathsf{T}} & \mathbf{D}\end{bmatrix}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \begin{bmatrix}\mathbf{e}_{\mathrm{p}}\\ \mathbf{e}_{\mathrm{X}}\end{bmatrix}$$

$$\longrightarrow \begin{bmatrix}\mathbf{I} & -\mathbf{B}\mathbf{D}^{-1}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\mathsf{T}} & \mathbf{D}\end{bmatrix}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \begin{bmatrix}\mathbf{I} & -\mathbf{B}\mathbf{D}^{-1}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{e}_{\mathrm{p}}\\ \mathbf{e}_{\mathrm{X}}\end{bmatrix}$$

$\mathbf{J}^{\mathsf{T}}\mathbf{J} =$

$$\mathbf{D} = \begin{bmatrix}\mathbf{d}_1 & & \\ & \ddots & \\ & & \mathbf{d}_M\end{bmatrix} \longrightarrow \mathbf{D}^{-1} = \begin{bmatrix}\mathbf{d}_1^{-1} & & \\ & \ddots & \\ & & \mathbf{d}_M^{-1}\end{bmatrix}$$

Inversion of block diagonal matrix can be efficiently computed.

$$\mathbf{J}^{\mathsf{T}}\mathbf{J} = \begin{bmatrix}\mathbf{J}_{\mathrm{p}}^{\mathsf{T}}\mathbf{J}_{\mathrm{p}} & \mathbf{J}_{\mathrm{p}}^{\mathsf{T}}\mathbf{J}_{\mathrm{X}}\\ \mathbf{J}_{\mathrm{X}}^{\mathsf{T}}\mathbf{J}_{\mathrm{p}} & \mathbf{J}_{\mathrm{X}}^{\mathsf{T}}\mathbf{J}_{\mathrm{X}}\end{bmatrix} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\mathsf{T}} & \mathbf{D}\end{bmatrix}$$

Normal equation:

$$\mathbf{J}^{\top}\mathbf{J}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \mathbf{J}^{\top}\left(\mathbf{b} - f(\mathbf{X})\right)$$

$$\longrightarrow \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\top} & \mathbf{D}\end{bmatrix}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \begin{bmatrix}\mathbf{e}_{\mathrm{p}}\\ \mathbf{e}_{\mathrm{X}}\end{bmatrix}$$

$$\longrightarrow \begin{bmatrix}\mathbf{I} & -\mathbf{B}\mathbf{D}^{-1}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\top} & \mathbf{D}\end{bmatrix}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \begin{bmatrix}\mathbf{I} & -\mathbf{B}\mathbf{D}^{-1}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{e}_{\mathrm{p}}\\ \mathbf{e}_{\mathrm{X}}\end{bmatrix}$$

$$\longrightarrow \begin{bmatrix}\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^{\top} & \mathbf{0}\\ \mathbf{B}^{\top} & \mathbf{D}\end{bmatrix}\begin{bmatrix}\Delta\mathbf{p}\\ \Delta\mathbf{X}\end{bmatrix} = \begin{bmatrix}\mathbf{e}_{\mathrm{p}} - \mathbf{B}\mathbf{D}^{-1}\mathbf{e}_{\mathrm{X}}\\ \mathbf{e}_{\mathrm{X}}\end{bmatrix}$$

$$\mathbf{J}^{\top}\mathbf{J} = \quad \mathbf{D} = \begin{bmatrix}\mathbf{d}_1 & & \\ & \ddots & \\ & & \mathbf{d}_M\end{bmatrix} \longrightarrow \mathbf{D}^{-1} = \begin{bmatrix}\mathbf{d}_1^{-1} & & \\ & \ddots & \\ & & \mathbf{d}_M^{-1}\end{bmatrix}$$

Inversion of block diagonal matrix can be efficiently computed.

$$\mathbf{J}^{\top}\mathbf{J} = \begin{bmatrix}\mathbf{J}_{\mathrm{p}}^{\top}\mathbf{J}_{\mathrm{p}} & \mathbf{J}_{\mathrm{p}}^{\top}\mathbf{J}_{\mathrm{X}}\\ \mathbf{J}_{\mathrm{X}}^{\top}\mathbf{J}_{\mathrm{p}} & \mathbf{J}_{\mathrm{X}}^{\top}\mathbf{J}_{\mathrm{X}}\end{bmatrix} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\top} & \mathbf{D}\end{bmatrix}$$

$$\mathbf{J}^\top\mathbf{J} = \quad \mathbf{D} = \begin{bmatrix} \mathbf{d}_1 & & \\ & \ddots & \\ & & \mathbf{d}_M \end{bmatrix} \longrightarrow \mathbf{D}^{-1} = \begin{bmatrix} \mathbf{d}_1^{-1} & & \\ & \ddots & \\ & & \mathbf{d}_M^{-1} \end{bmatrix}$$

Inversion of block diagonal matrix can be efficiently computed.

$$\mathbf{J}^\top\mathbf{J} = \begin{bmatrix} \mathbf{J}_p^\top\mathbf{J}_p & \mathbf{J}_p^\top\mathbf{J}_X \\ \mathbf{J}_X^\top\mathbf{J}_p & \mathbf{J}_X^\top\mathbf{J}_X \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{D} \end{bmatrix}$$

Normal equation:

$$\mathbf{J}^\top\mathbf{J}\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{X} \end{bmatrix} = \mathbf{J}^\top\left(\mathbf{b} - f(\mathbf{X})\right)$$

$$\longrightarrow \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{D} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

$$\longrightarrow \begin{bmatrix} \mathbf{I} & -\mathbf{B}\mathbf{D}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{D} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & -\mathbf{B}\mathbf{D}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_X \end{bmatrix}$$

$$\longrightarrow \begin{bmatrix} \mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^\top & \mathbf{0} \\ \mathbf{B}^\top & \mathbf{D} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_p - \mathbf{B}\mathbf{D}^{-1}\mathbf{e}_X \\ \mathbf{e}_X \end{bmatrix}$$

$$\longrightarrow \Delta\mathbf{p} = \left(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^\top\right)^{-1}\left(\mathbf{e}_p - \mathbf{B}\mathbf{D}^{-1}\mathbf{e}_X\right)$$

$$\Delta\mathbf{X} = \mathbf{D}^{-1}\left(\mathbf{e}_X - \mathbf{B}^\top\Delta\mathbf{p}\right)$$

Note: $\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^\top$ is Schur complement of $\mathbf{D}$

$$J^T J = \quad D = \begin{bmatrix} d_1 & & \\ & \ddots & \\ & & d_M \end{bmatrix} \longrightarrow D^{-1} = \begin{bmatrix} d_1^{-1} & & \\ & \ddots & \\ & & d_M^{-1} \end{bmatrix}$$

Inversion of block diagonal matrix can be efficiently computed.

$$J^T J = \begin{bmatrix} J_p^T J_p & J_p^T J_X \\ J_X^T J_p & J_X^T J_X \end{bmatrix} = \begin{bmatrix} A & B \\ B^T & D \end{bmatrix}$$

Normal equation:

$$J^T J \begin{bmatrix} \Delta p \\ \Delta X \end{bmatrix} = J^T \left(b - f(X) \right)$$

$$\longrightarrow \begin{bmatrix} A & B \\ B^T & D \end{bmatrix} \begin{bmatrix} \Delta p \\ \Delta X \end{bmatrix} = \begin{bmatrix} e_p \\ e_X \end{bmatrix}$$

$$\longrightarrow \begin{bmatrix} I & -BD^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} A & B \\ B^T & D \end{bmatrix} \begin{bmatrix} \Delta p \\ \Delta X \end{bmatrix} = \begin{bmatrix} I & -BD^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} e_p \\ e_X \end{bmatrix}$$

$$\longrightarrow \begin{bmatrix} A - BD^{-1}B^T & 0 \\ B^T & D \end{bmatrix} \begin{bmatrix} \Delta p \\ \Delta X \end{bmatrix} = \begin{bmatrix} e_p - BD^{-1}e_X \\ e_X \end{bmatrix}$$

Small size matrix

$$\longrightarrow \Delta p = \boxed{\left(A - BD^{-1}B^T \right)^{-1}} \left(e_p - BD^{-1}e_X \right)$$

$$\Delta X = D^{-1} \left(e_X - B^T \Delta p \right)$$

Note: $A - BD^{-1}B^T$ is Schur complement of D


SIFT detection
Reprojection


SIFT detection
Reprojection

Measurement
Linear estimate (reproj: 0.199104)
Nonlinear estimate (reproj: 0.119272)


Measurement
Linear estimate (reproj: 0.199104)
Nonlinear estimate (reproj: 0.119272)