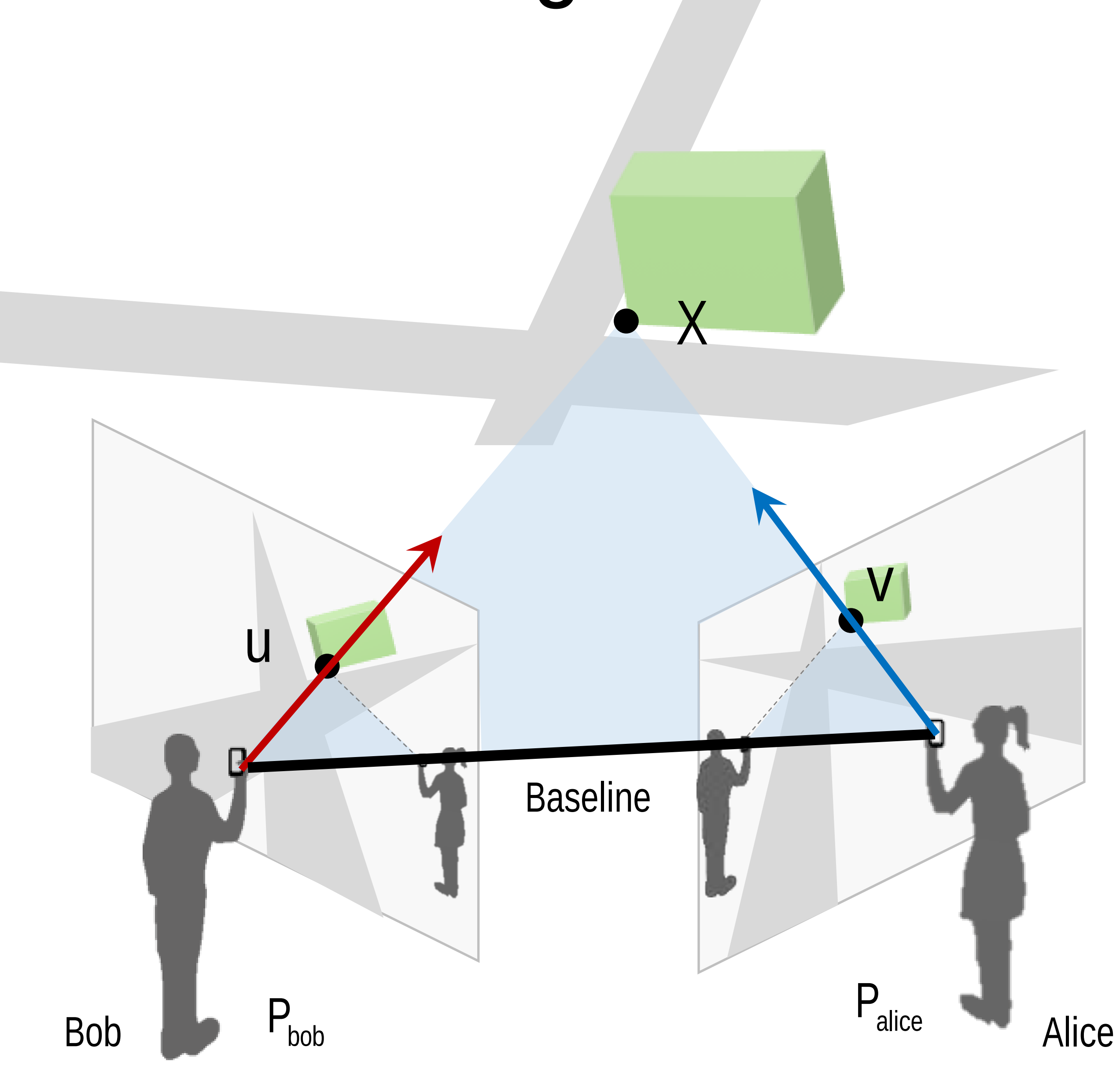
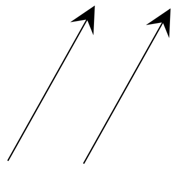
Triangulation Refinement

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} u \\ 1 \end{bmatrix} = P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix}$$



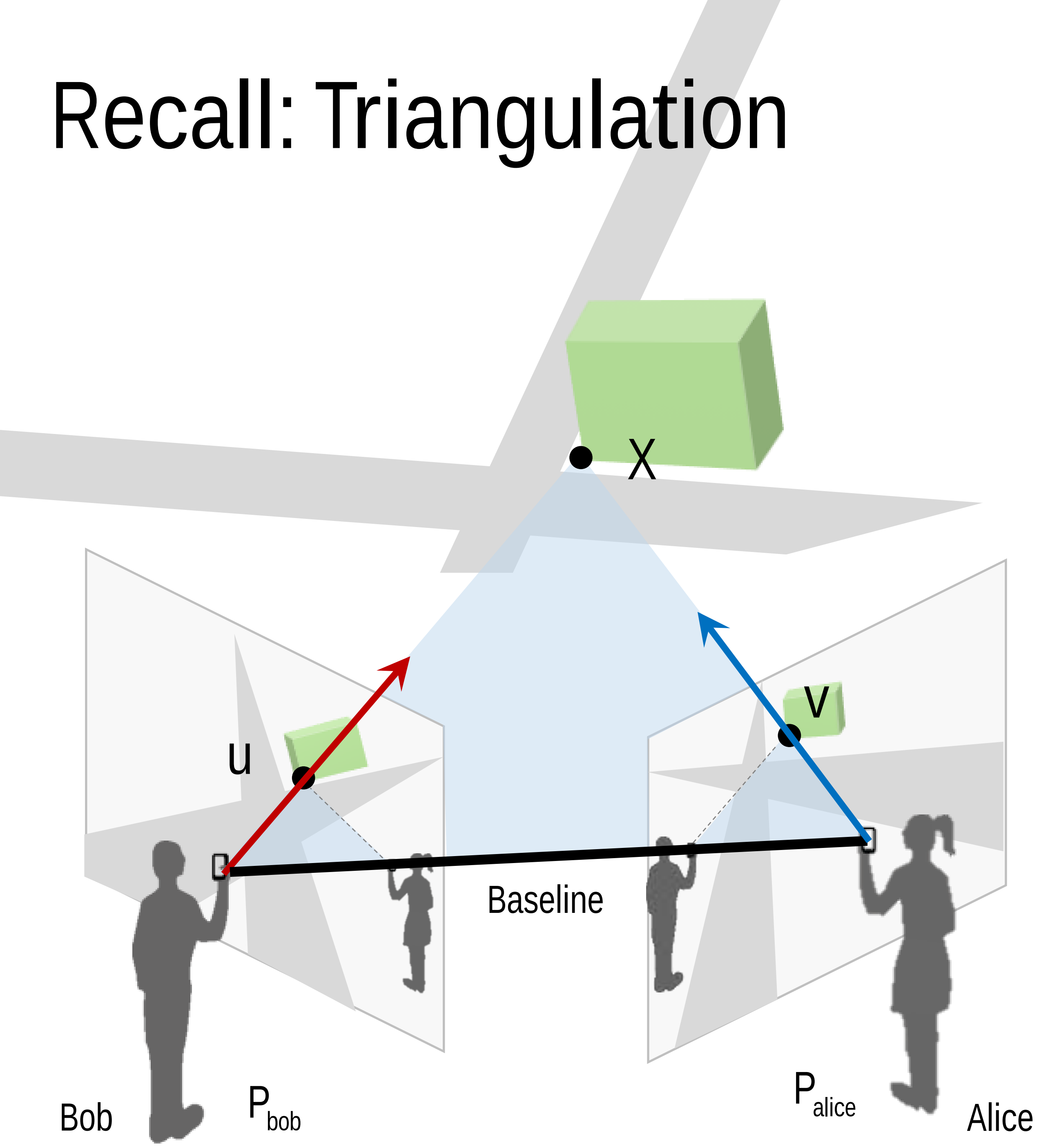
Two 3D vectors are parallel.

$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix} \times P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

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

Skew-symmetric matrix

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} u \\ 1 \end{bmatrix} = P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix}$$

Two 3D vectors are parallel.

$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix} \times P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

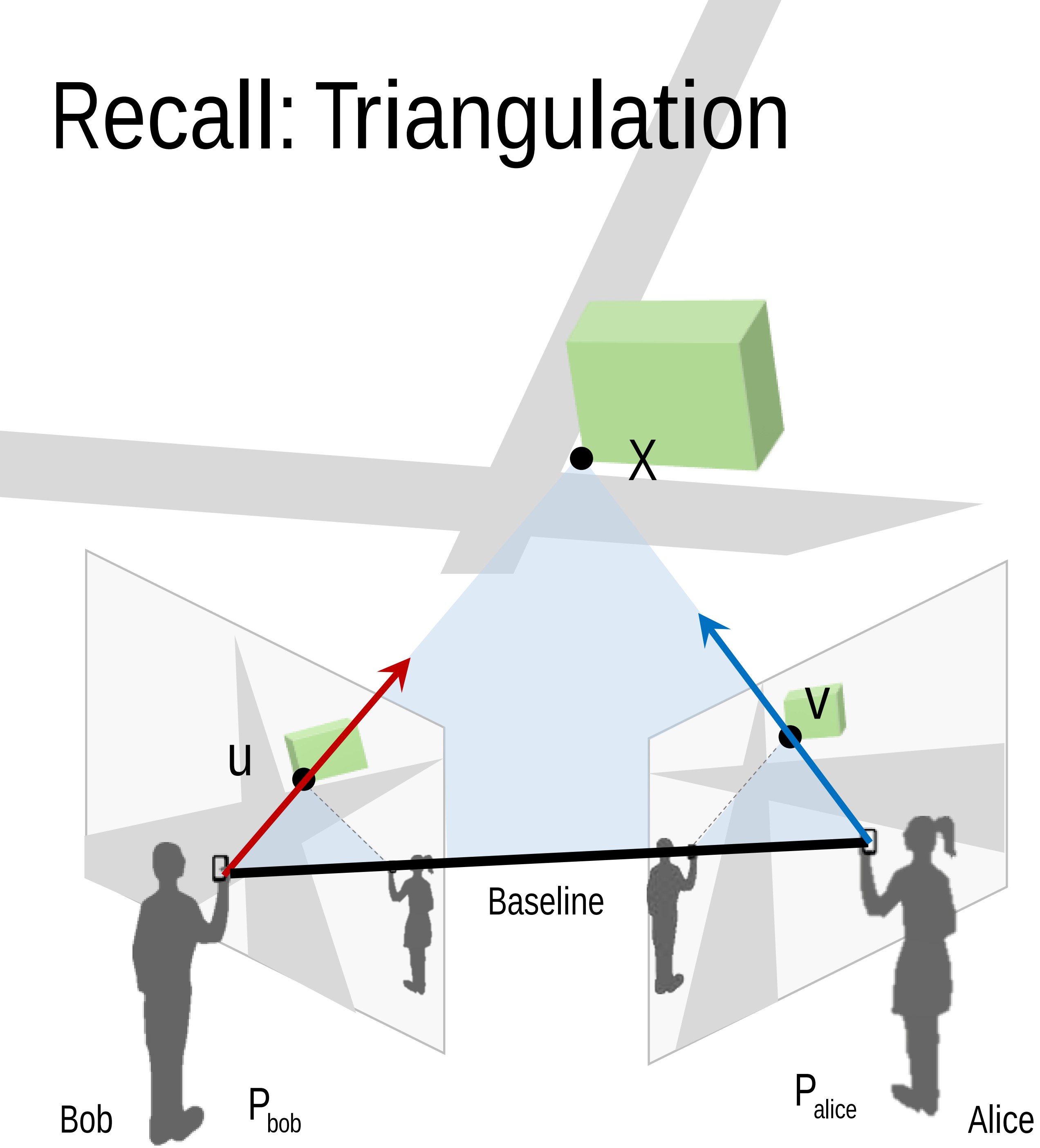
$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix}_{\times} P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

Skew-symmetric matrix

: Knowns

: Unknowns

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} u \\ 1 \end{bmatrix} = P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix}$$

Two 3D vectors are parallel.

$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix} \times P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

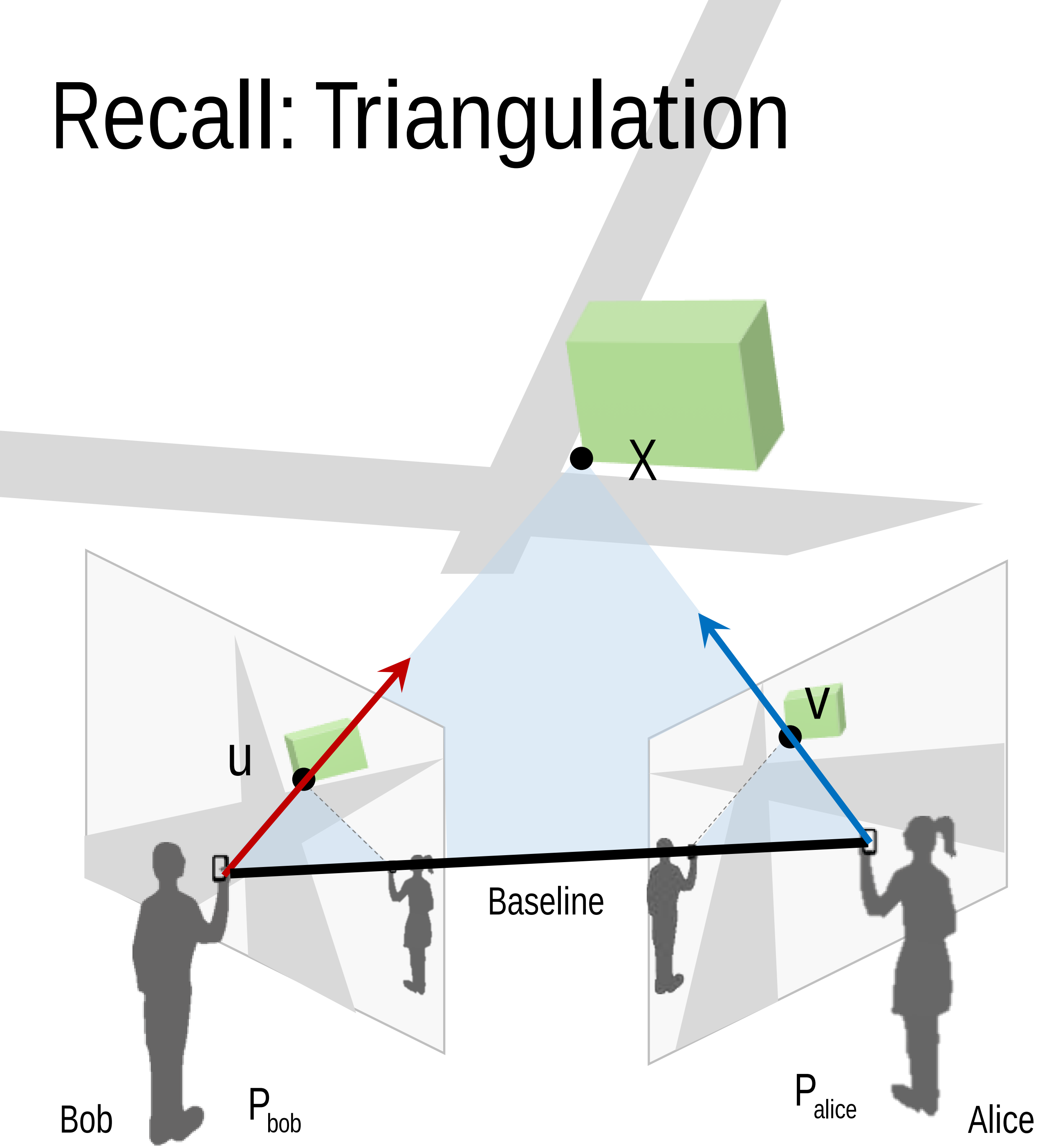
$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix}_{\times} P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

2x4

: Knowns

: Unknowns

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} u \\ 1 \end{bmatrix} = P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix}$$

Two 3D vectors are parallel.

$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix} \times P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

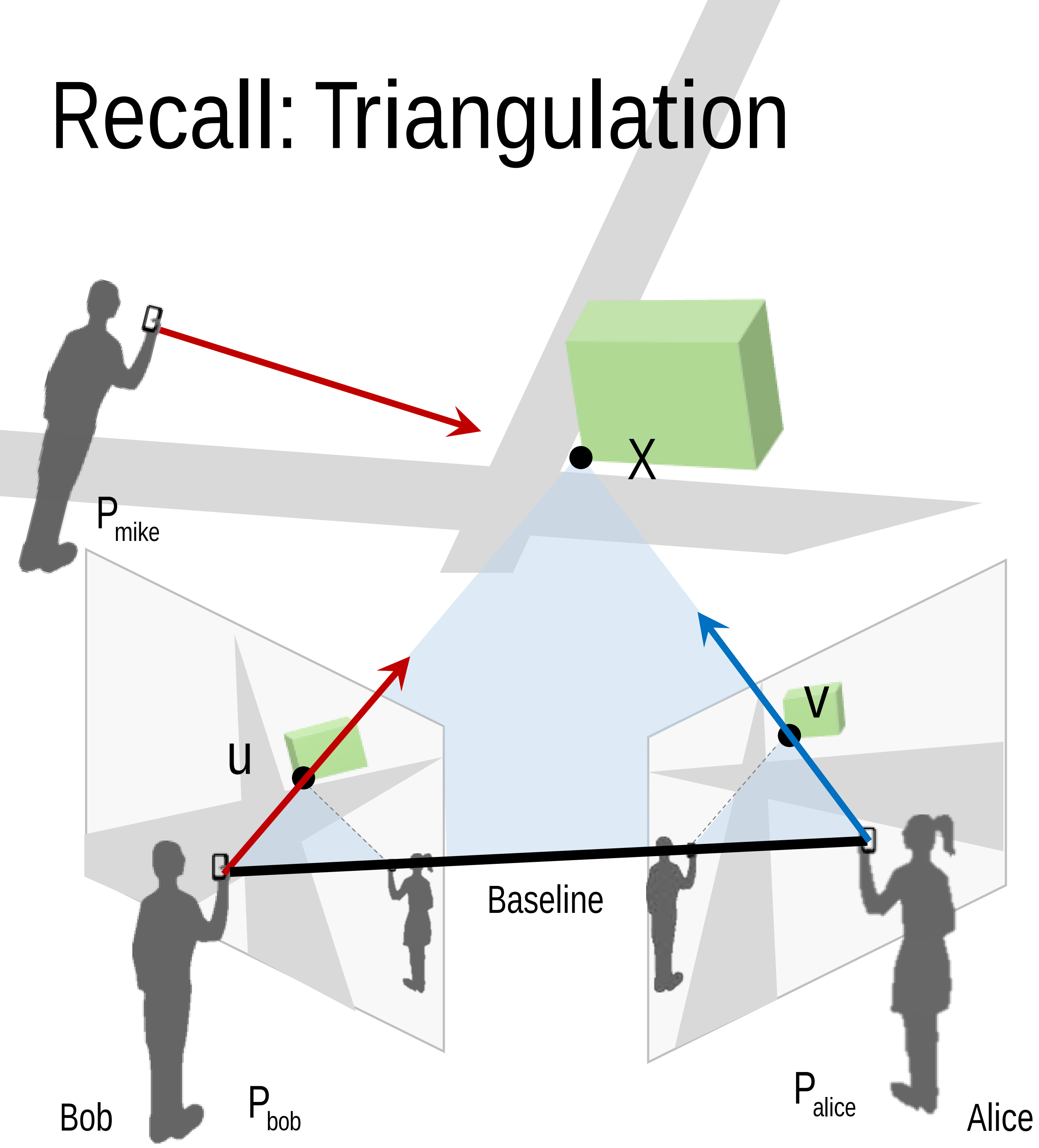
$$\rightarrow \begin{bmatrix} \begin{bmatrix} u \\ 1 \end{bmatrix}_{\times} P_{bob} \\ \begin{bmatrix} v \\ 1 \end{bmatrix}_{\times} P_{alice} \end{bmatrix} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

4x4

: Knowns

: Unknowns

Recall: Triangulation



General camera pose

$$\lambda_1 \begin{bmatrix} u \\ 1 \end{bmatrix} = P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix}$$

Two 3D vectors are parallel.

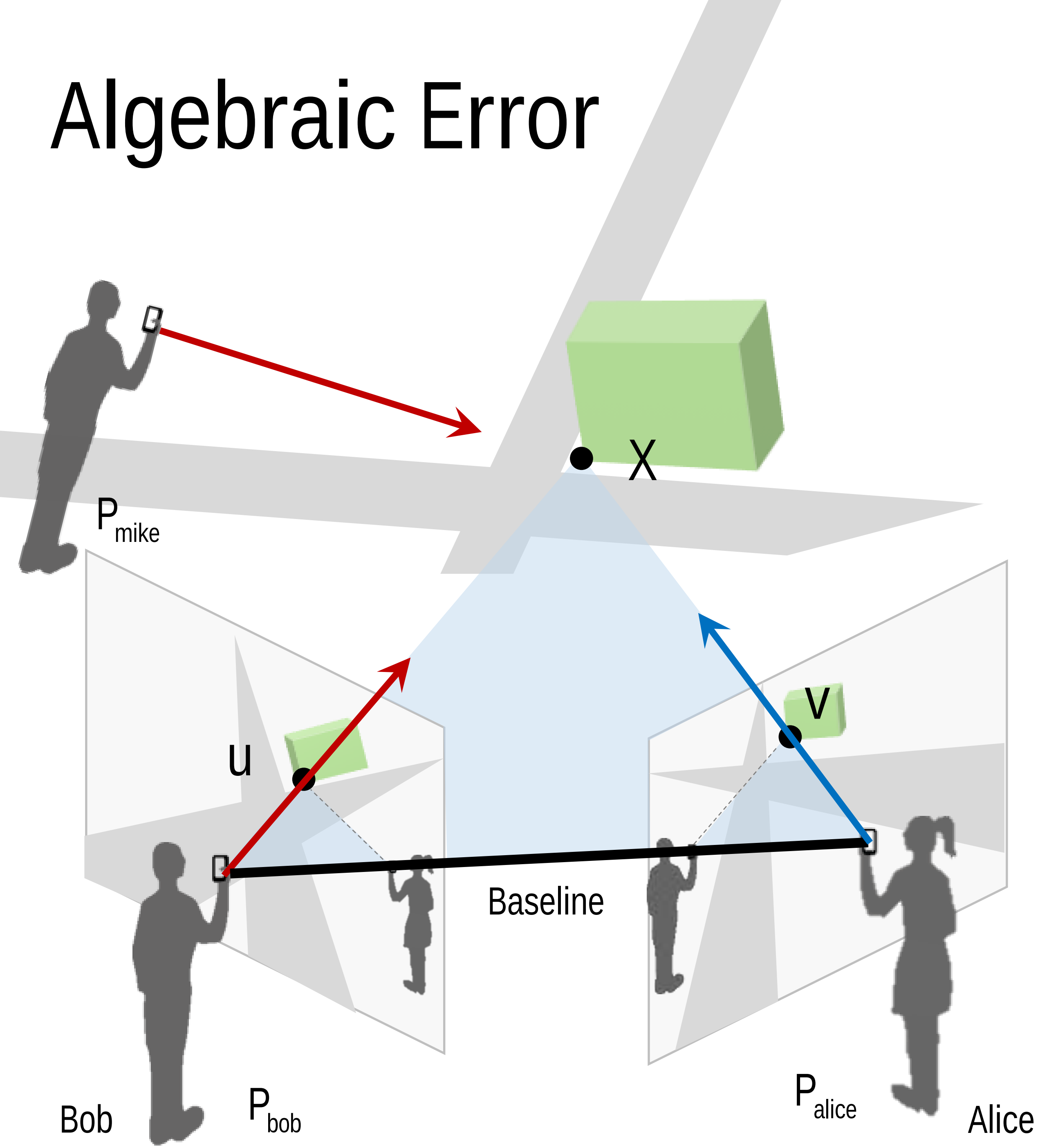
$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix} \times P_{bob} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

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

: Knowns

: Unknowns

Algebraic Error



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

Algebraic error does not have geometric meaning.

Recall: Geometric Verification via Reprojection Error

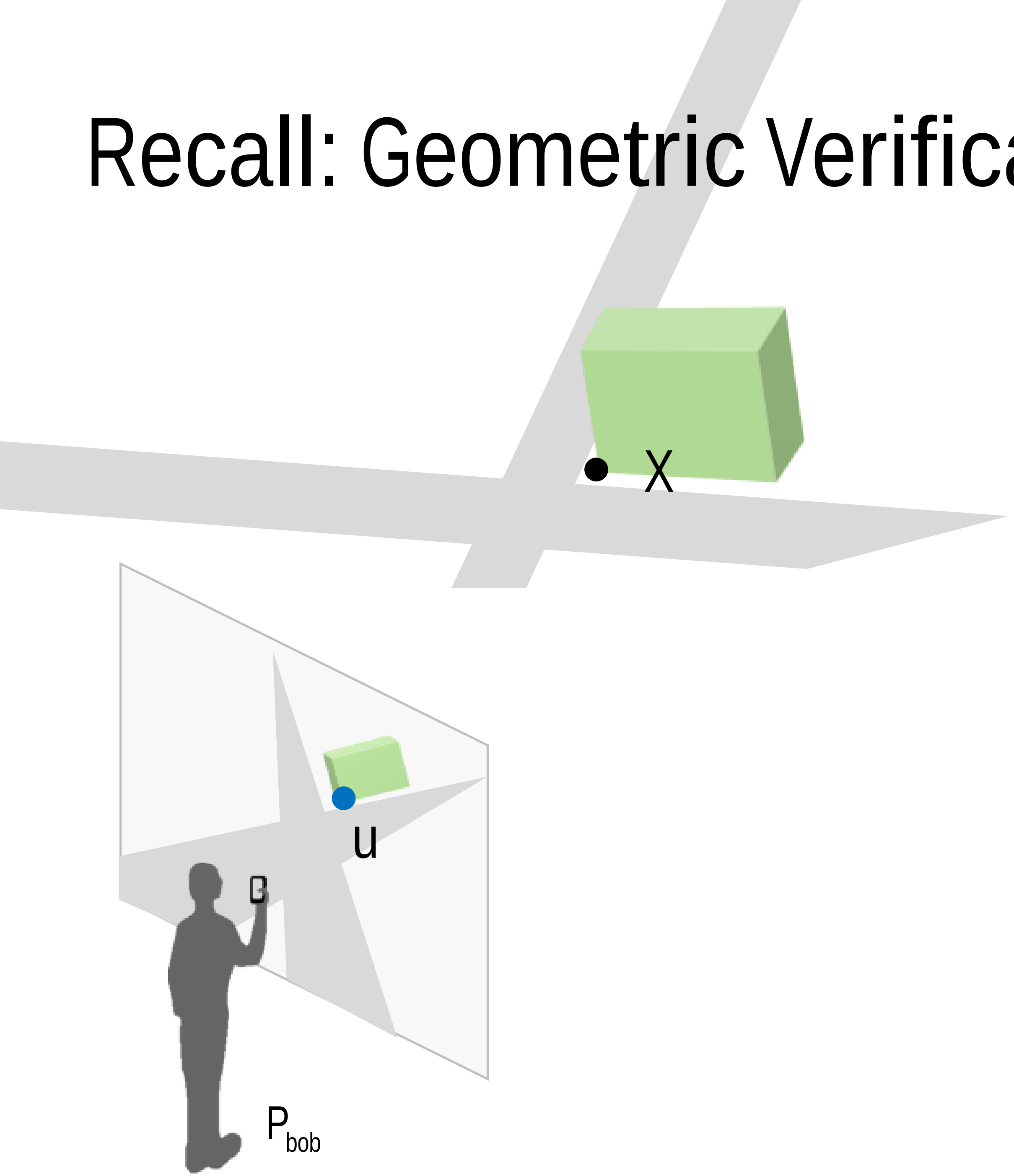


Image feature measurement, e.g. SIFT detection:

u

Recall: Geometric Verification via Reprojection Error

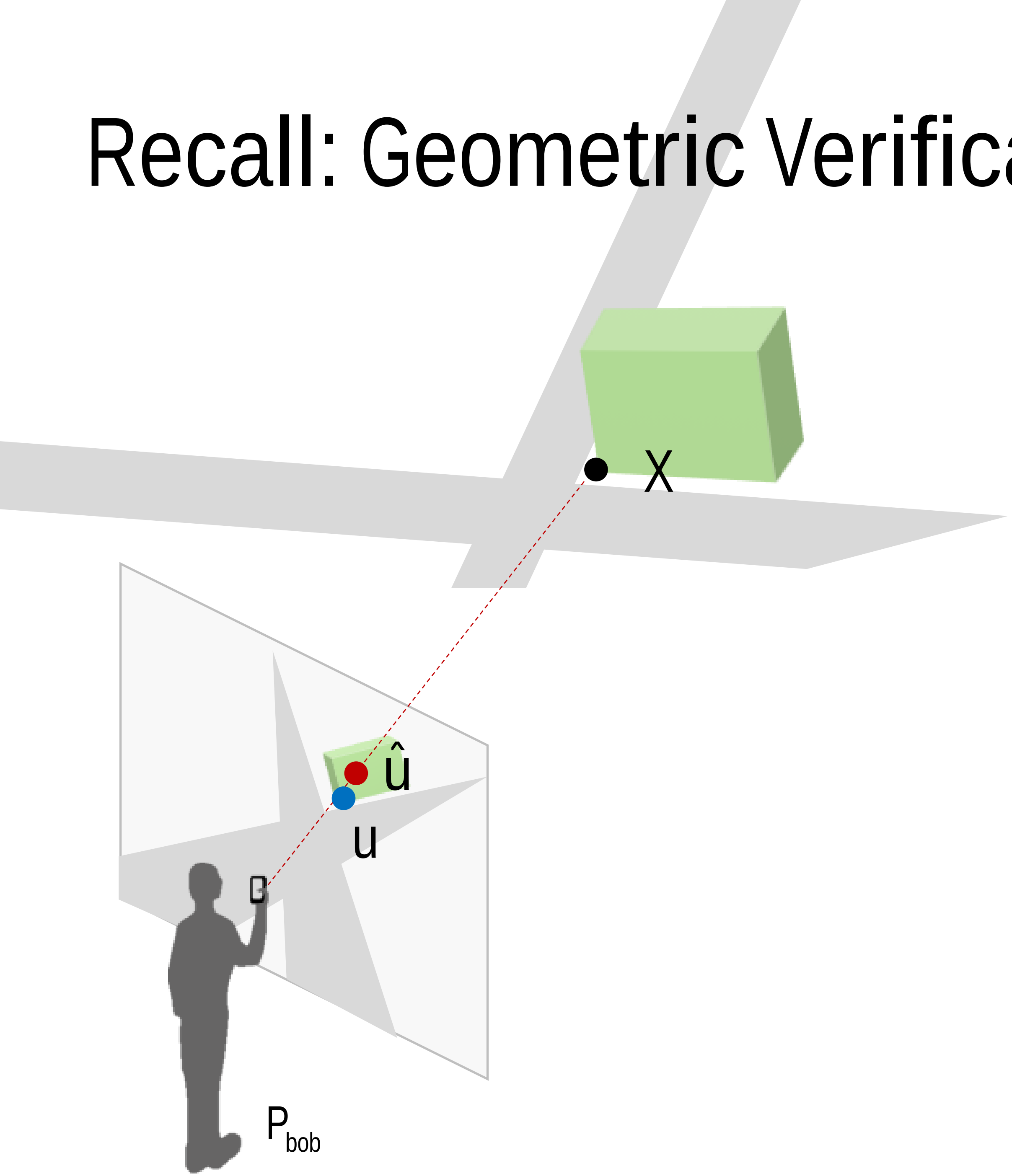


Image feature measurement, e.g. SIFT detection:

$\mathbf{u}$

3D point projection, or reprojection:

$$\lambda \hat{u} = PX$$

Recall: Geometric Verification via Reprojection Error

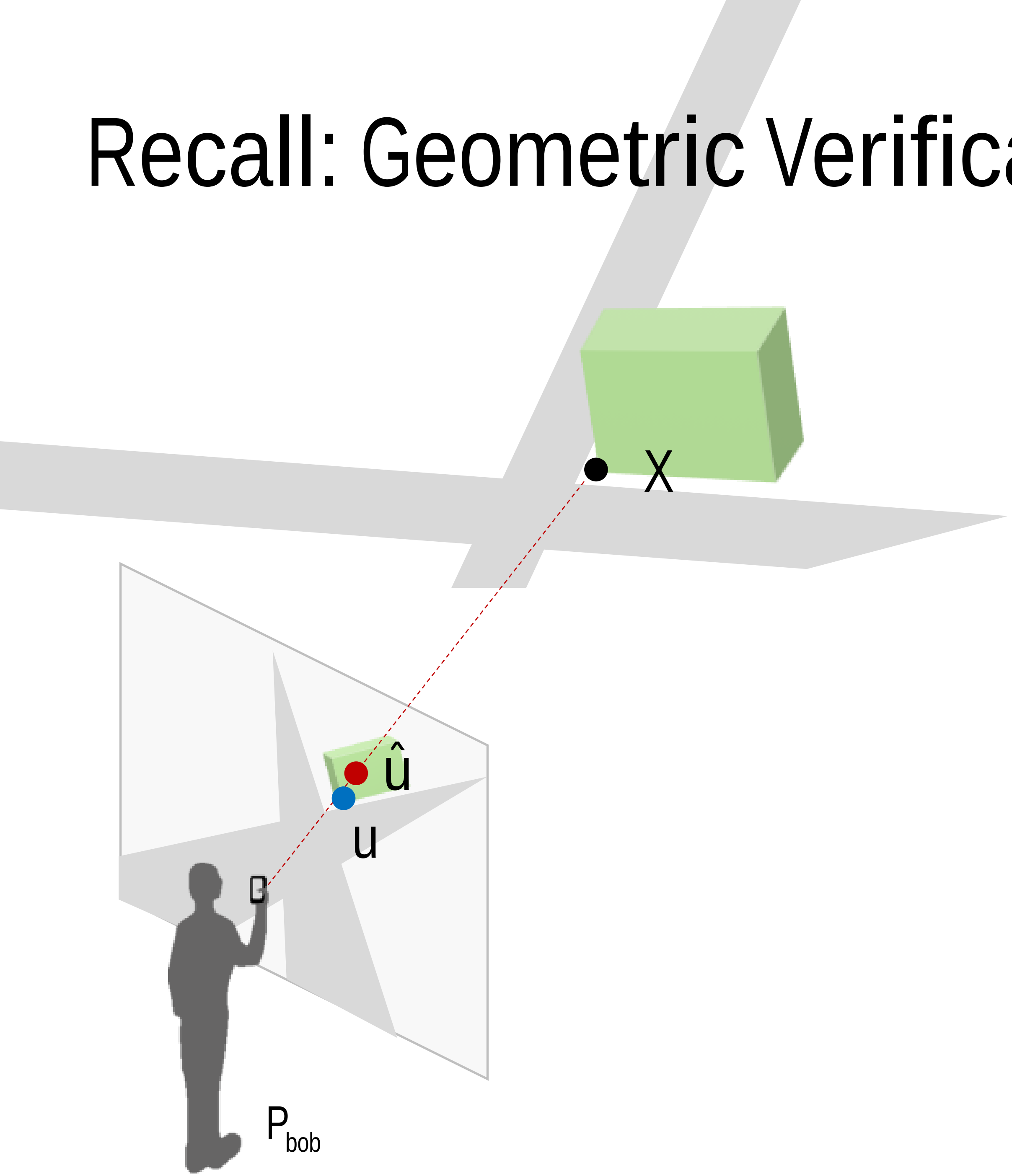


Image feature measurement, e.g. SIFT detection:

$$\mathrm{u}$$

3D point projection, or reprojection:

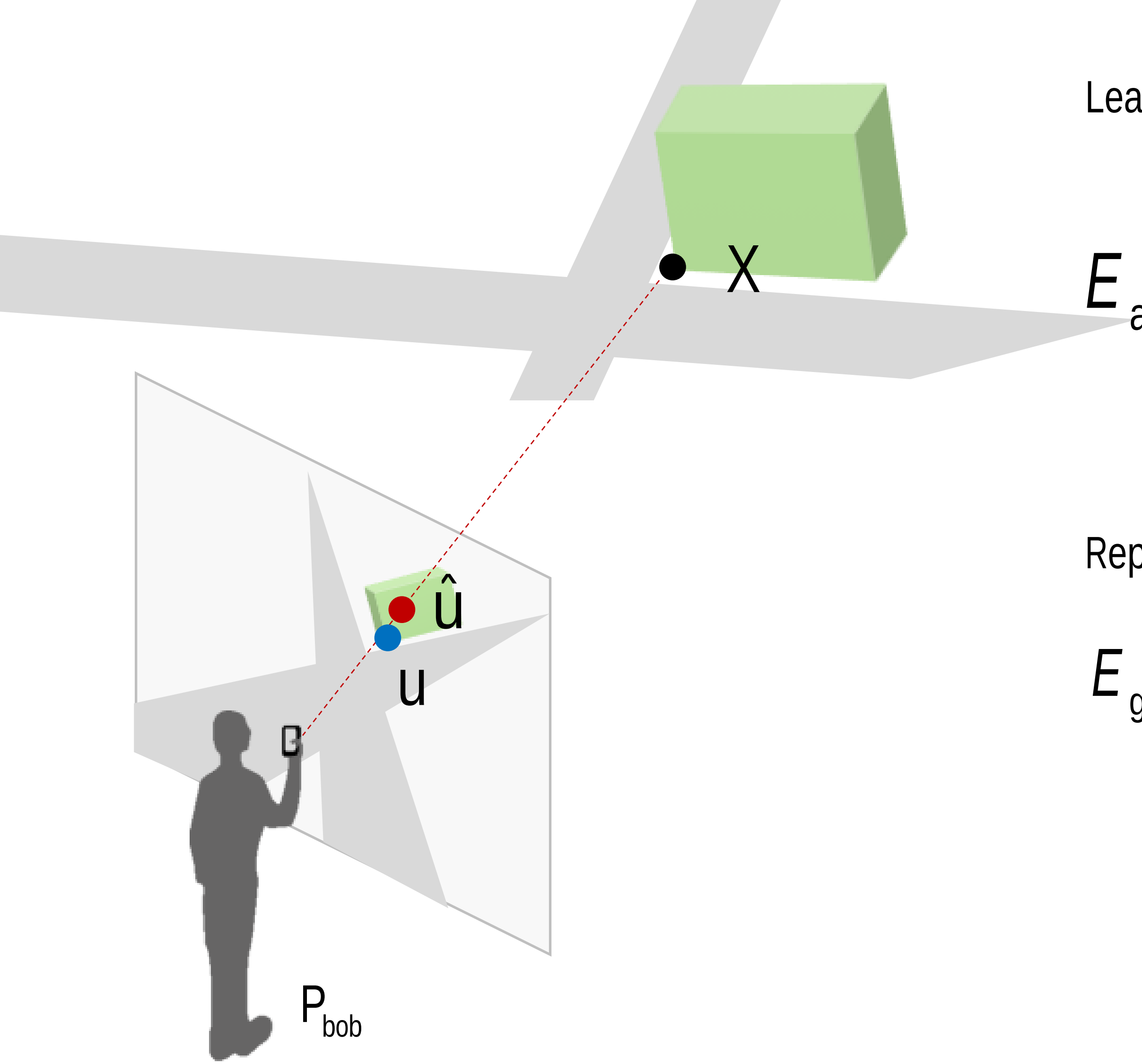
$$\lambda \hat{\mathrm{u}} = \mathrm{PX}$$

Reprojection error (geometric error):

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

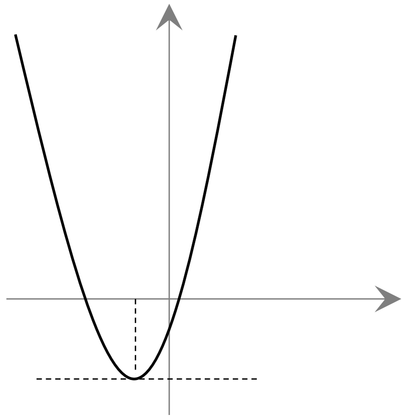
$$= \left(\frac{\mathrm{P}_1\mathrm{X}}{\mathrm{P}_3\mathrm{X}} - \mathrm{u}_1 \right)^2 + \left(\frac{\mathrm{P}_2\mathrm{X}}{\mathrm{P}_3\mathrm{X}} - \mathrm{u}_2 \right)^2$$

Algebraic vs. Geometric error



Least squares solution (algebraic error):

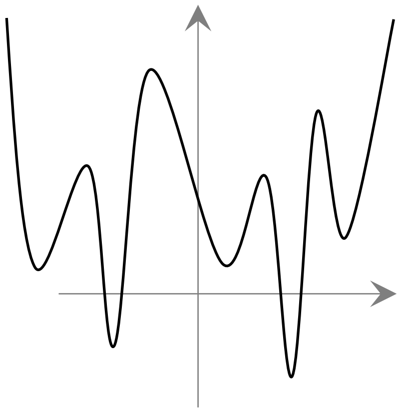
$$E_{alge} = \| A x - b \|^2$$



Reprojection error (geometric error):

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

$$= \left(\frac{P_1 X}{P_3 X} - u_1\right)^2 + \left(\frac{P_2 X}{P_3 X} - u_2\right)^2$$

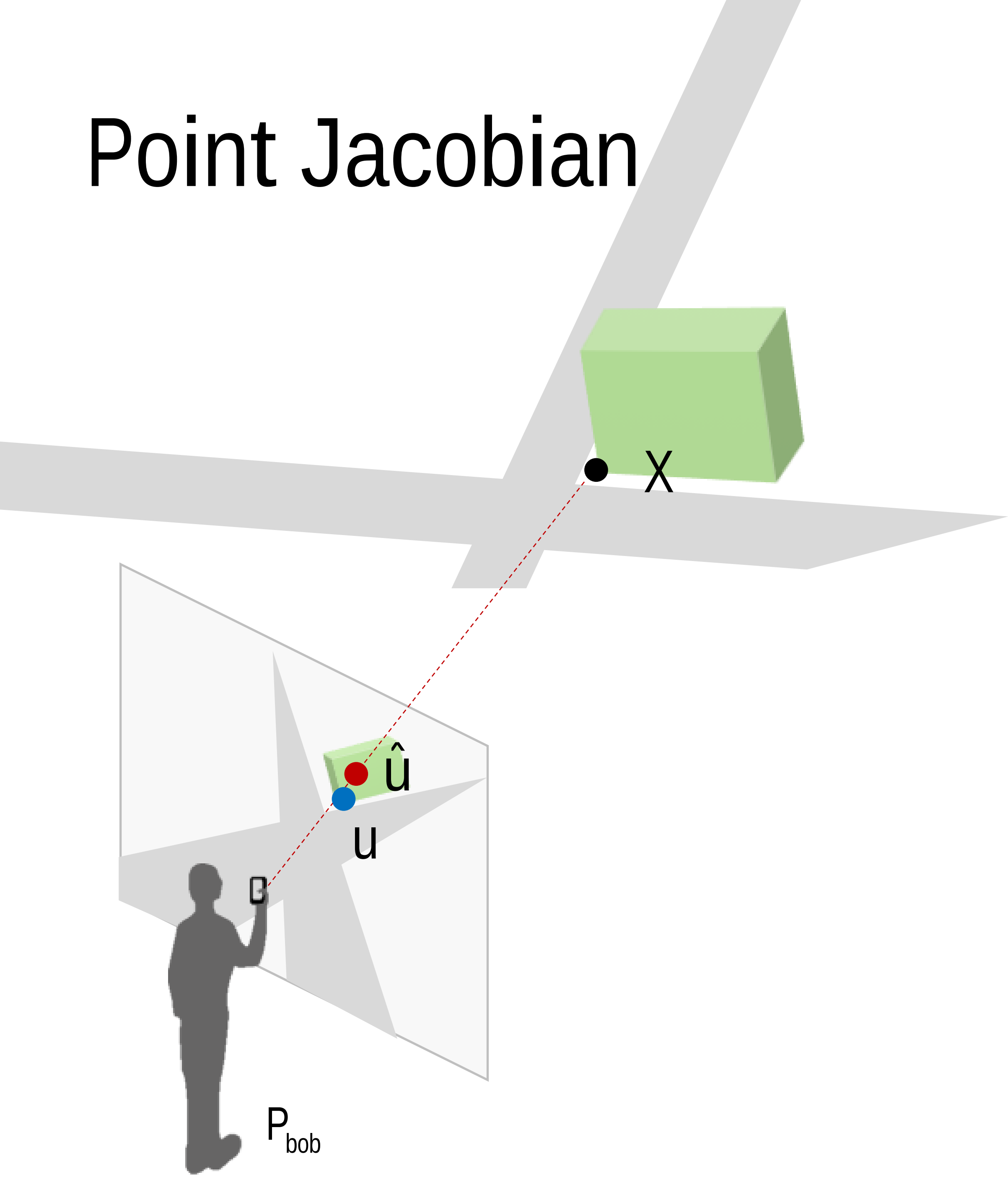


Point Jacobian

Black: given variables
Red: unknowns

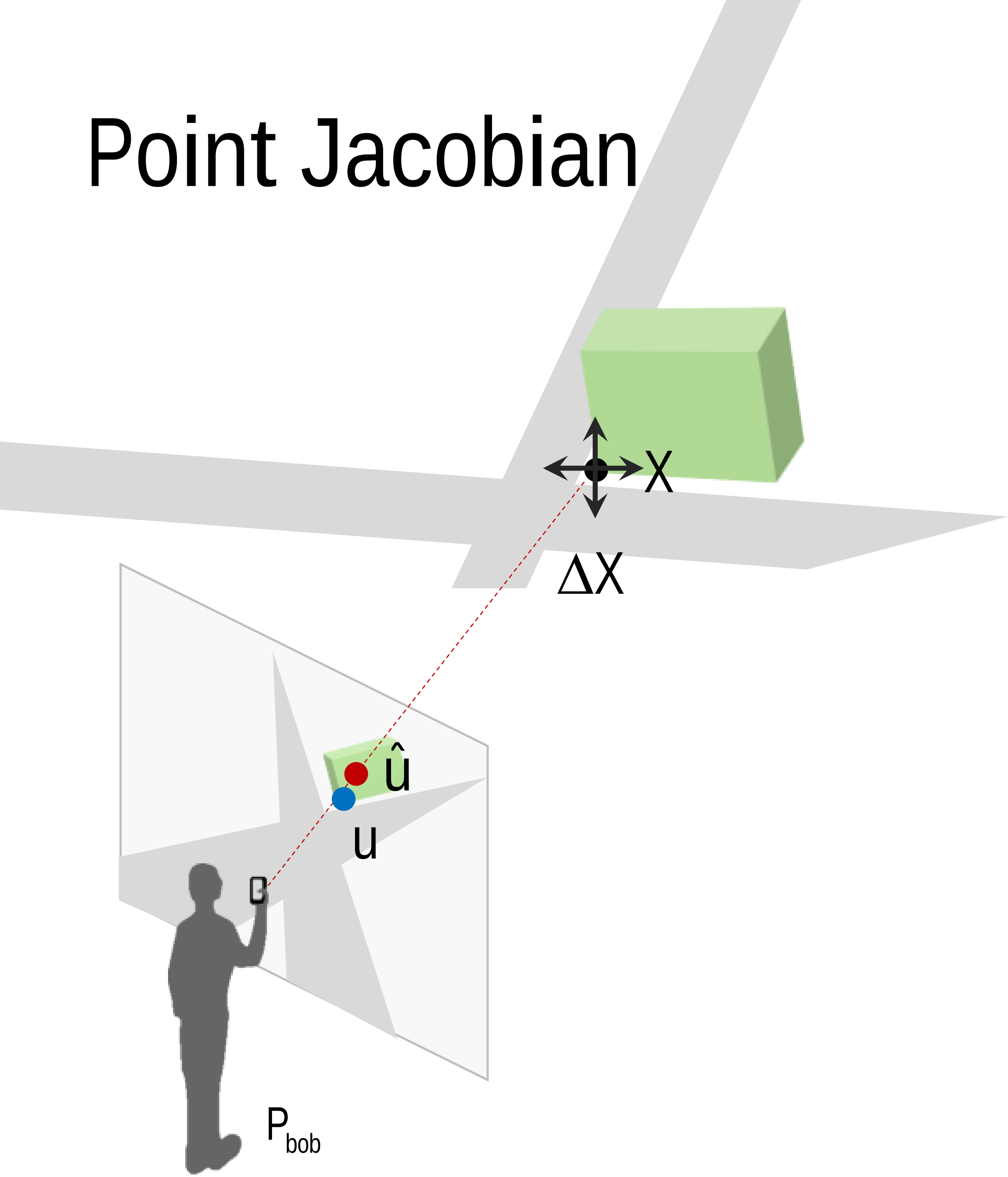
$$E_{geom} = \left\| \hat{\mathbf{u}} - \mathbf{u} \right\|^2$$

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



Point Jacobian

Black: given variables
Red: unknowns



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

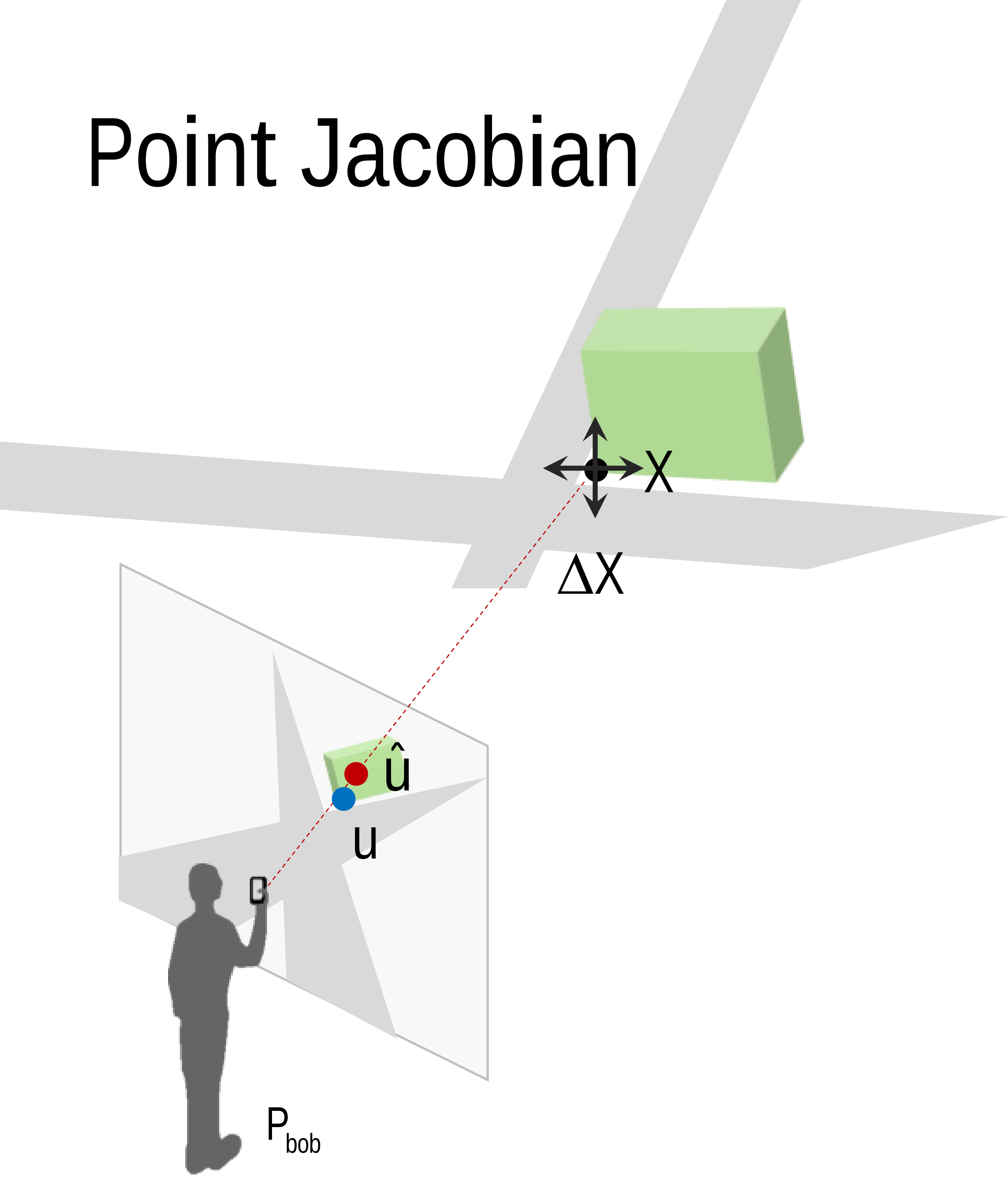
$$= \left(\frac{P_1 X}{P_3 X} - x \right)^2 + \left(\frac{P_2 X}{P_3 X} - y \right)^2$$

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

Point Jacobian

Black: given variables
Red: unknowns

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

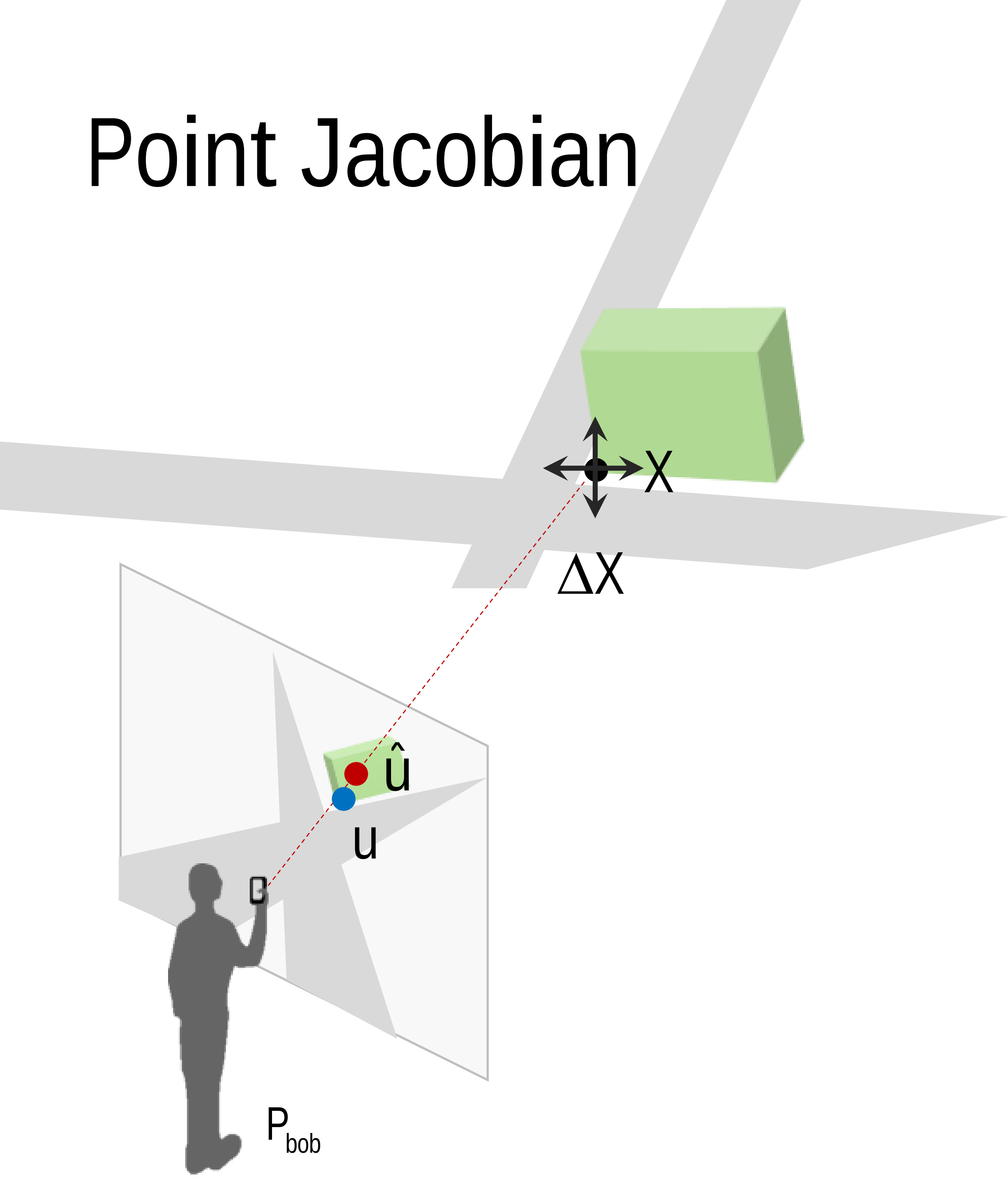


$$\Delta x = \left(\frac{\partial f(x)^T}{\partial x}\frac{\partial f(x)}{\partial x}\right)^{-1}\frac{\partial f(x)^T}{\partial x}(b - f(x))$$

Point Jacobian

Black: given variables
Red: unknowns

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



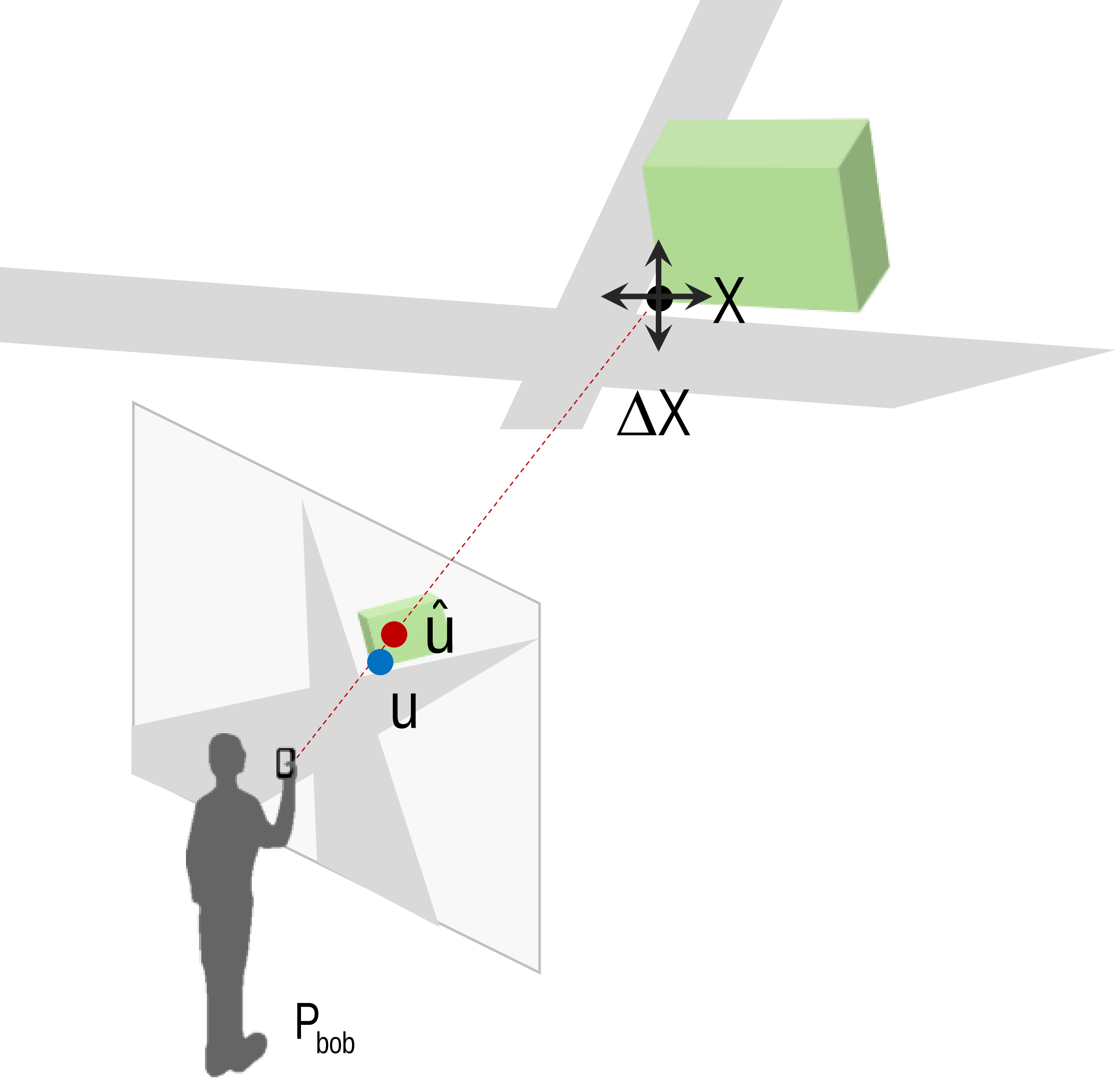
$$f(X) = \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix}$$

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

Point Jacobian

Black: given variables
Red: unknowns

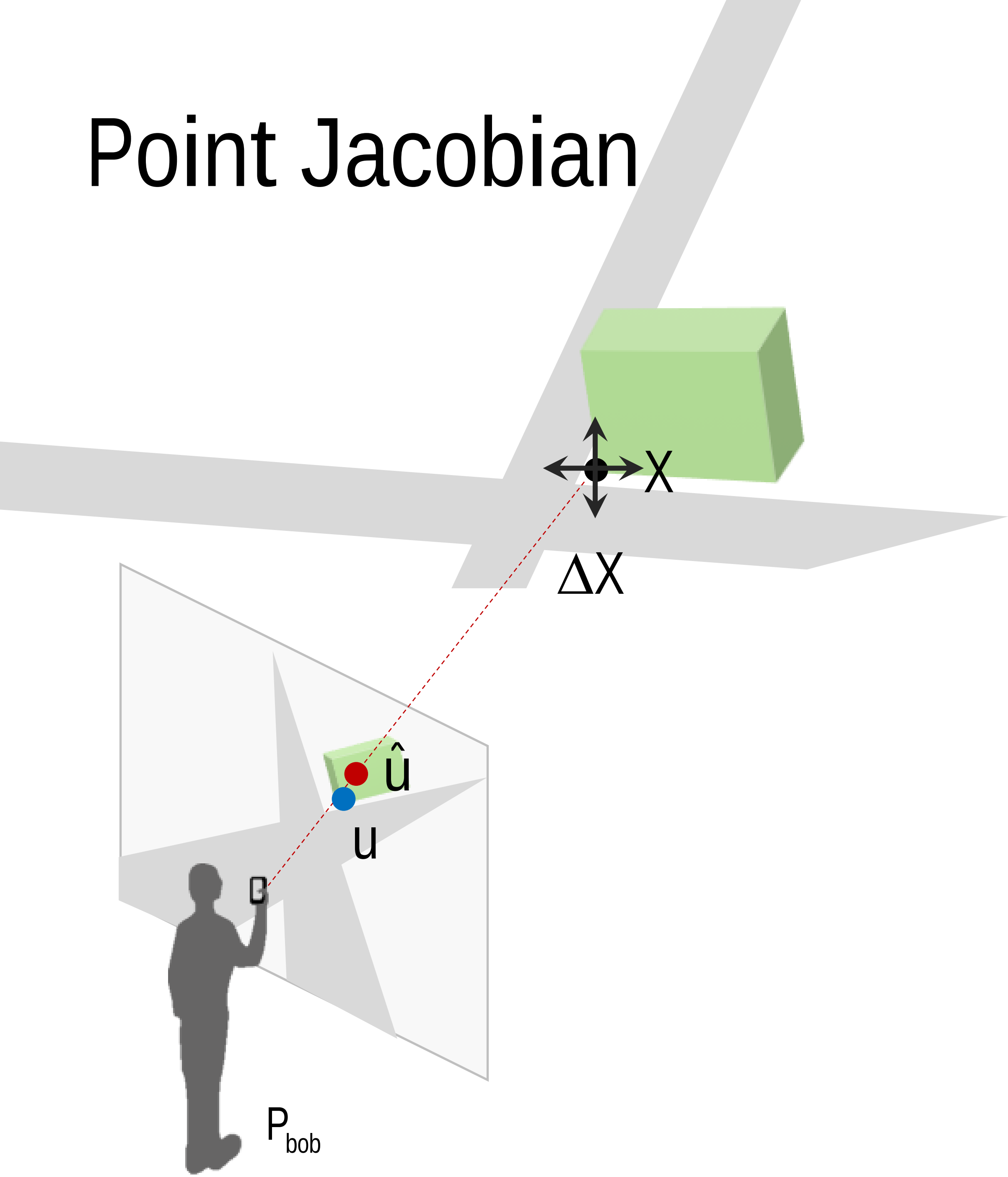
$$E_{geom} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = KR(X - C)$$



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

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

Point Jacobian



Black: given variables

Red: unknowns

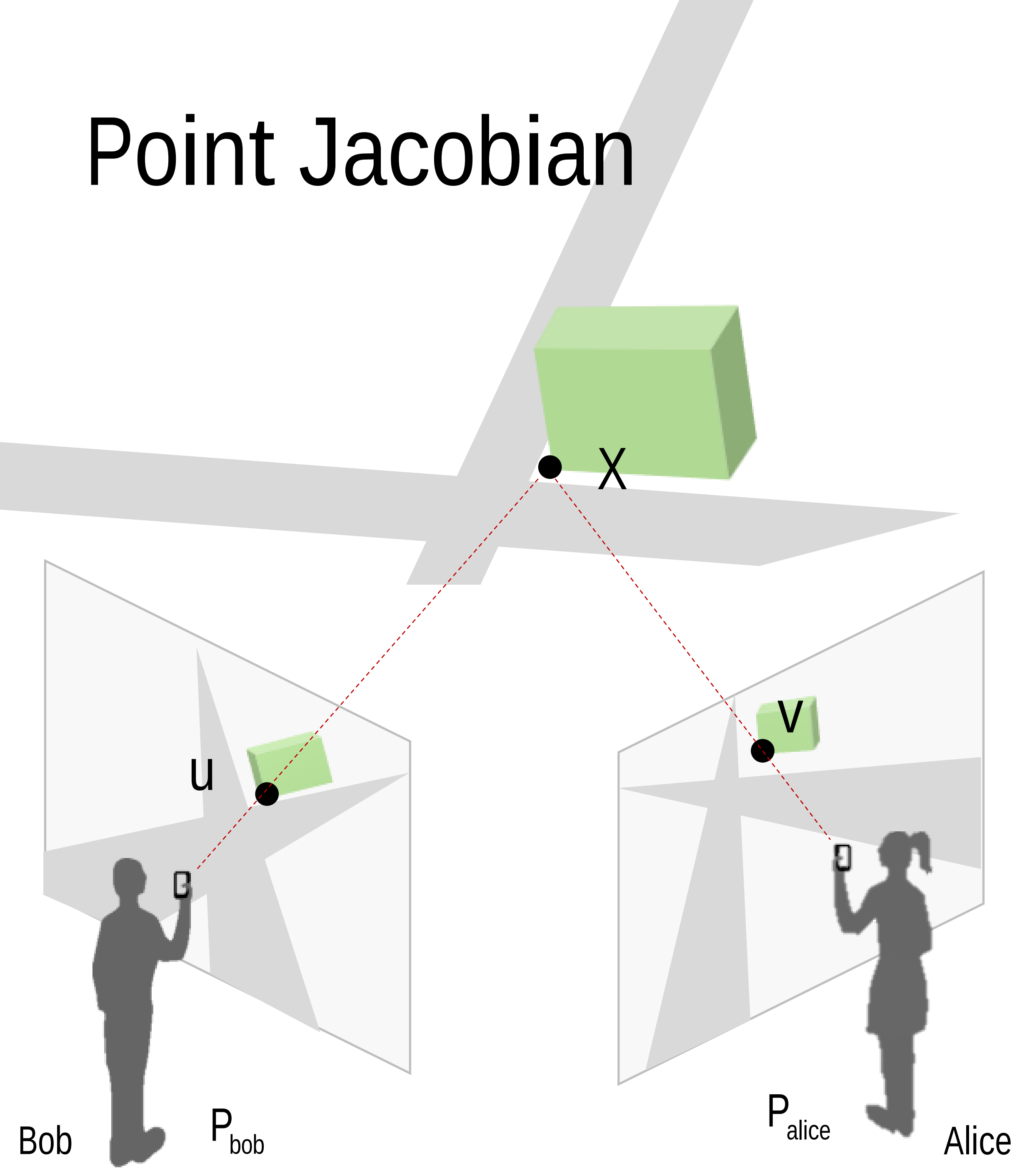
$$E_{geom} = \left(\frac{u}{w} - x\right)^2 + \left(\frac{v}{w} - y\right)^2 \quad \text{where} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = KR(X - C)$$

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

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

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

Point Jacobian



Black: given variables
Red: unknowns

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

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

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

Algorithm 3 Nonlinear Point Refinement

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

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

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

Algorithm 3 Nonlinear Point Refinement

$\mathbf{b} = [\ \mathbf{u}_1^\mathsf{T} \ \ \mathbf{u}_2^\mathsf{T}\ ]^\mathsf{T}$
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

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

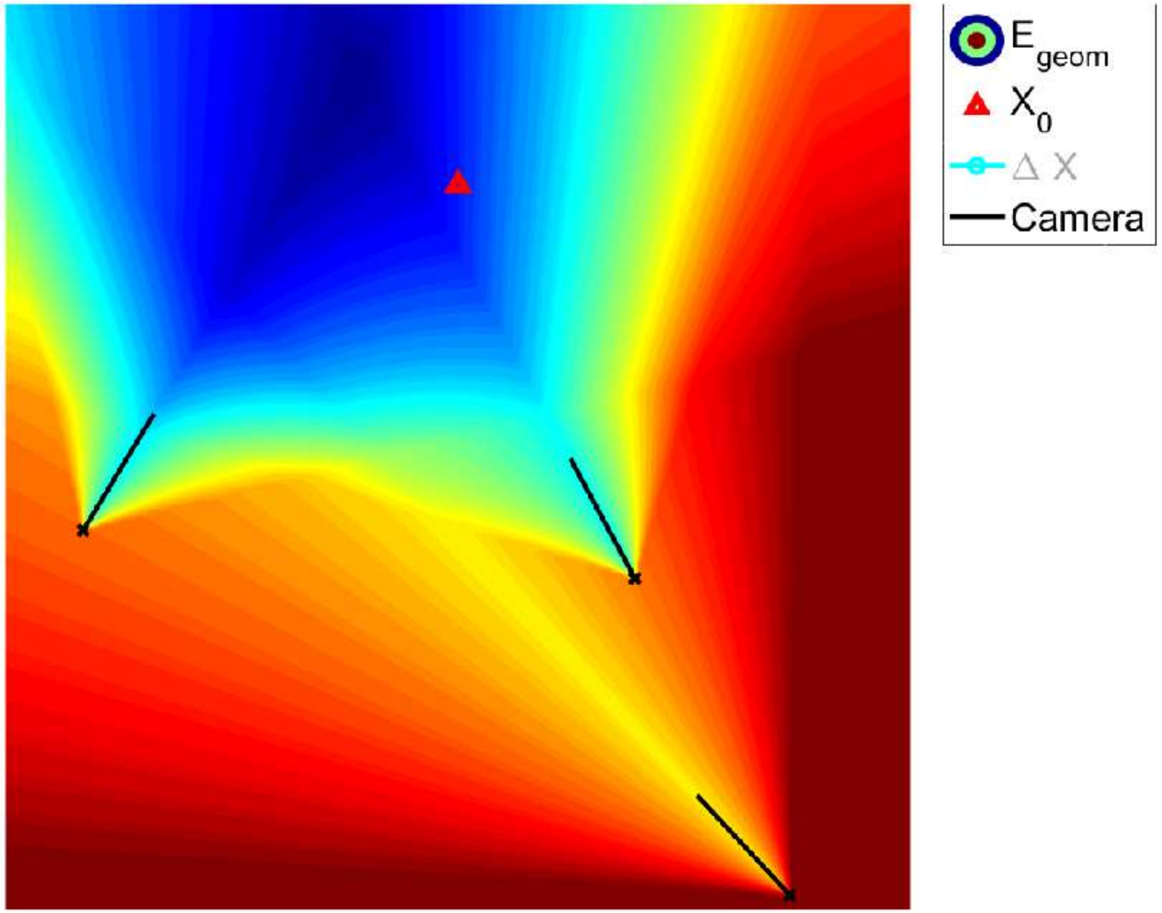
$$\Delta x = \left(\frac{\partial f(x)}{\partial x}^{T} \frac{\partial f(x)}{\partial x} \right)^{-1} \frac{\partial f(x)}{\partial x}^{T} (b - f(x))$$

Damping factor (Levenberg-Marquardt algorithm)

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

Black: given variables
Red: unknowns

Example: 1D Camera Triangulation

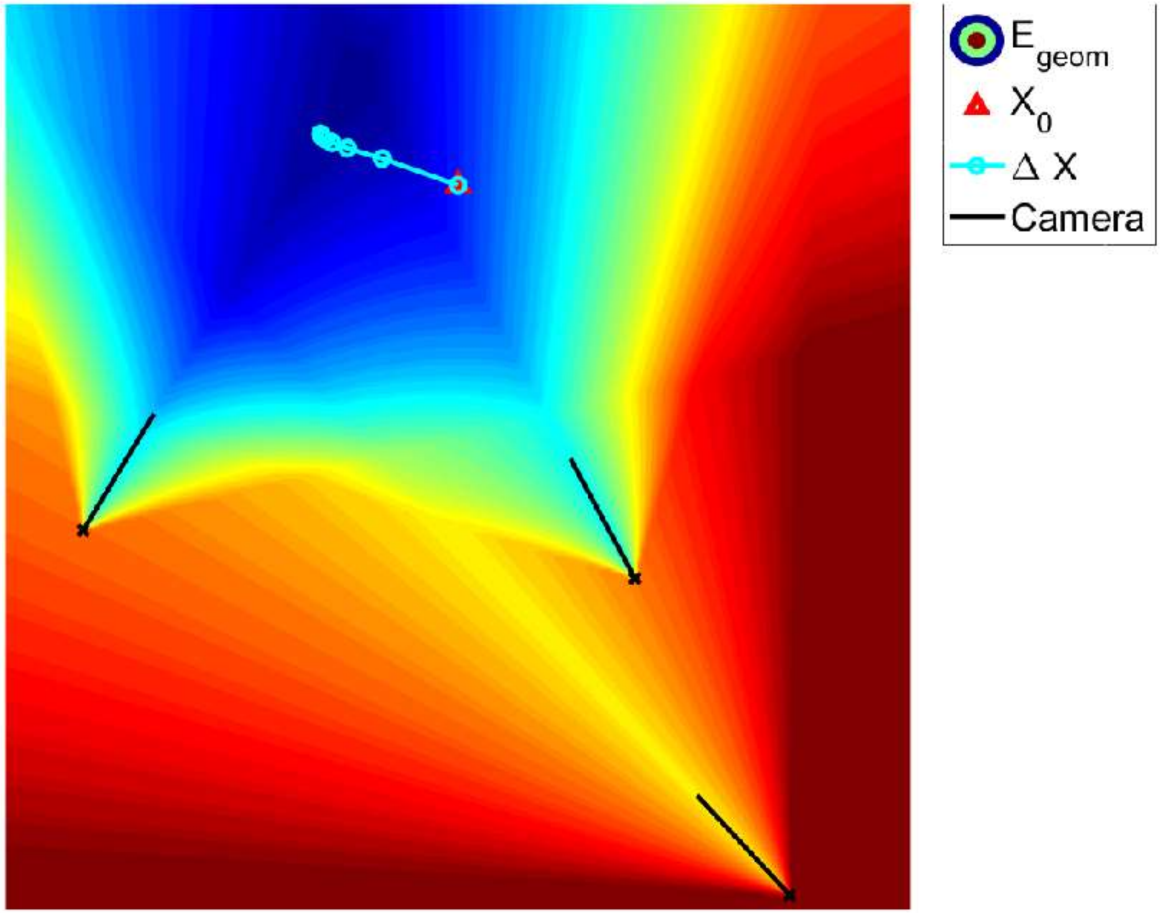



```
function df_dx = JacobianX_1D(K, R,  C,  X)

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];
```

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

Black: given variables
Red: unknowns

Example: 1D Camera Triangulation

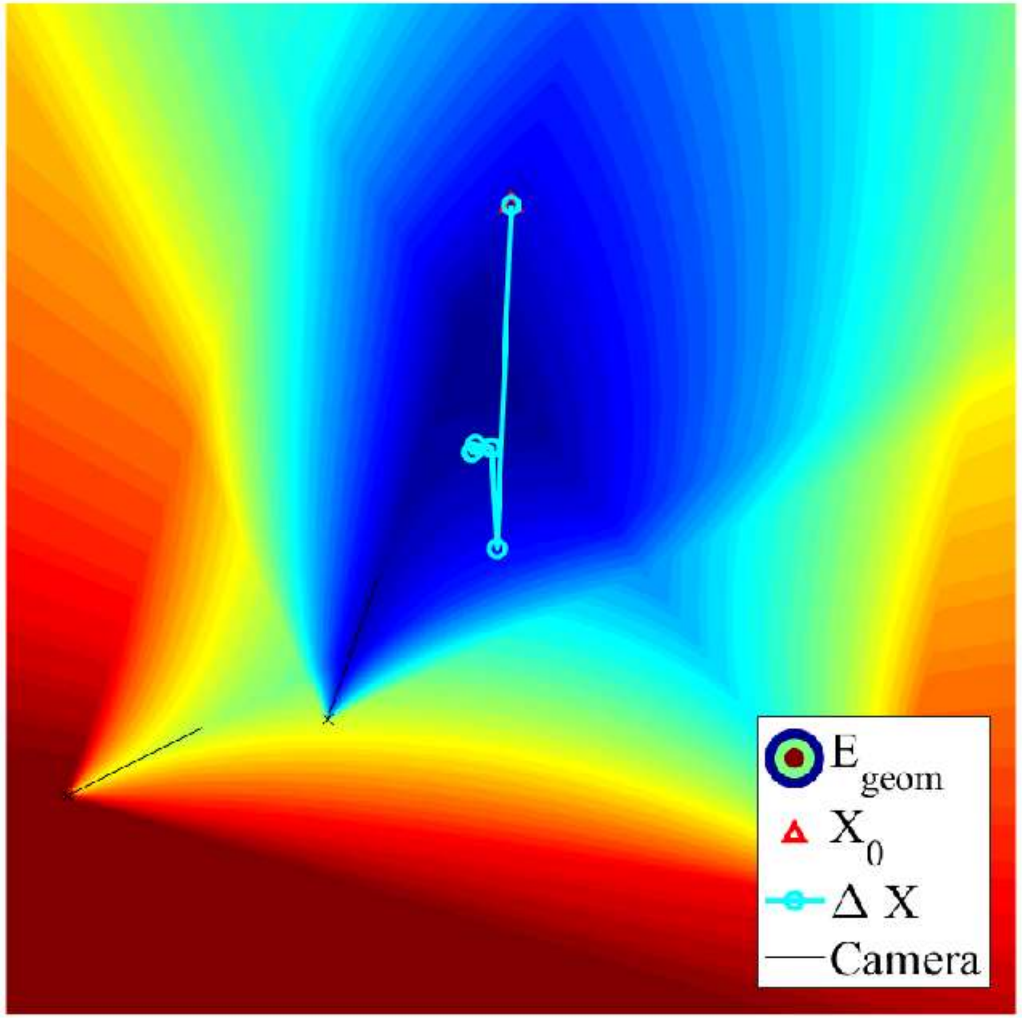


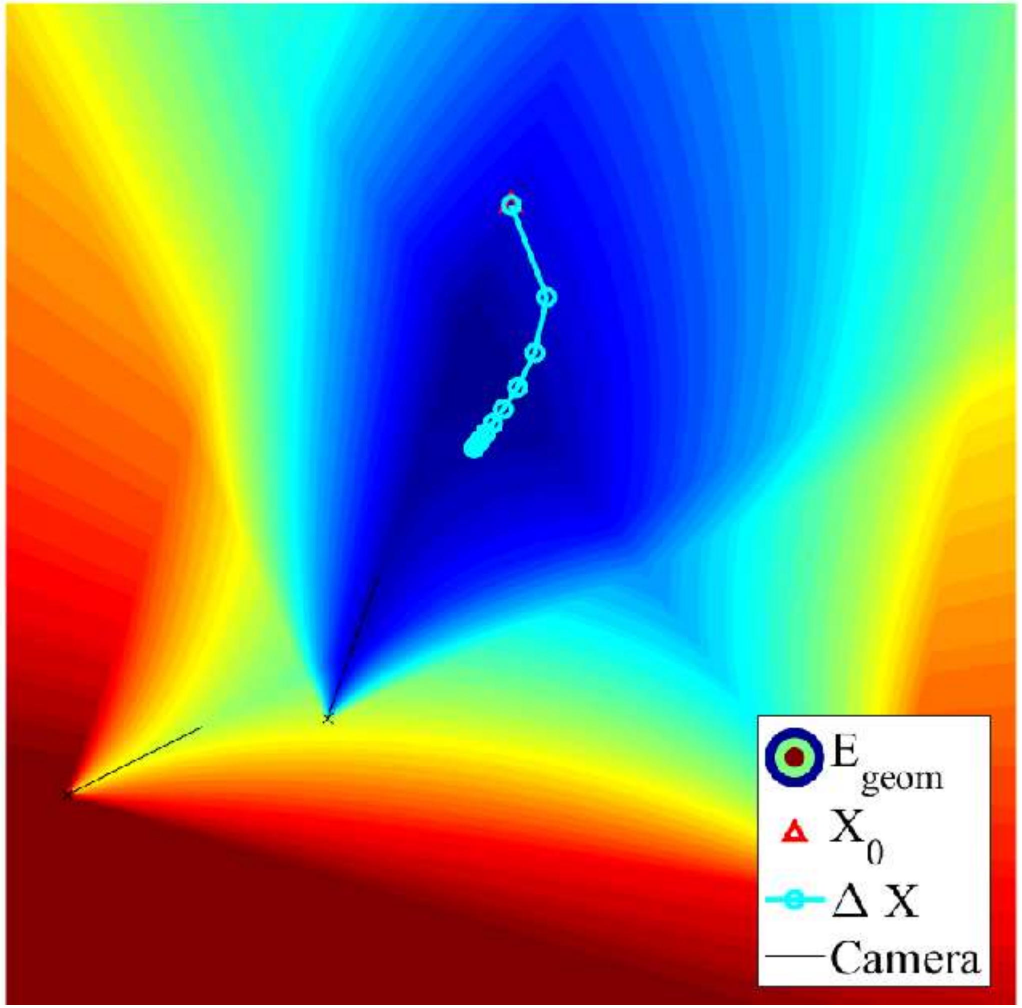
NonlinearTriangulation1D.m

```
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
```

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

Damping Factor



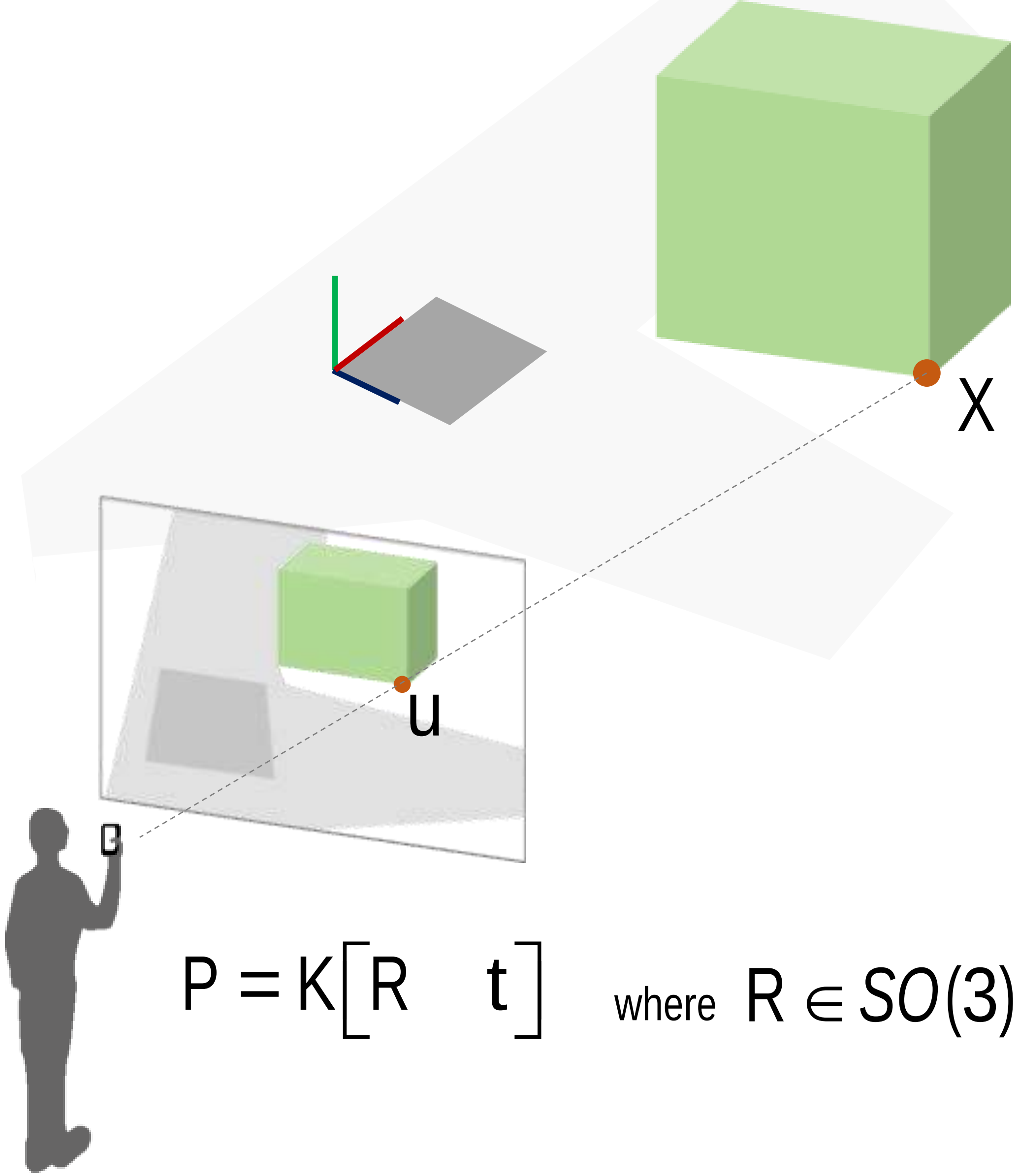


$$\Delta x = \left(\frac{\partial f(x)}{\partial x}^{T} \frac{\partial f(x)}{\partial x} \right)^{-1} \frac{\partial f(x)}{\partial x}^{T} \left(b - f(x) \right)$$

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

PnP Refinement

3D-2D Correspondence



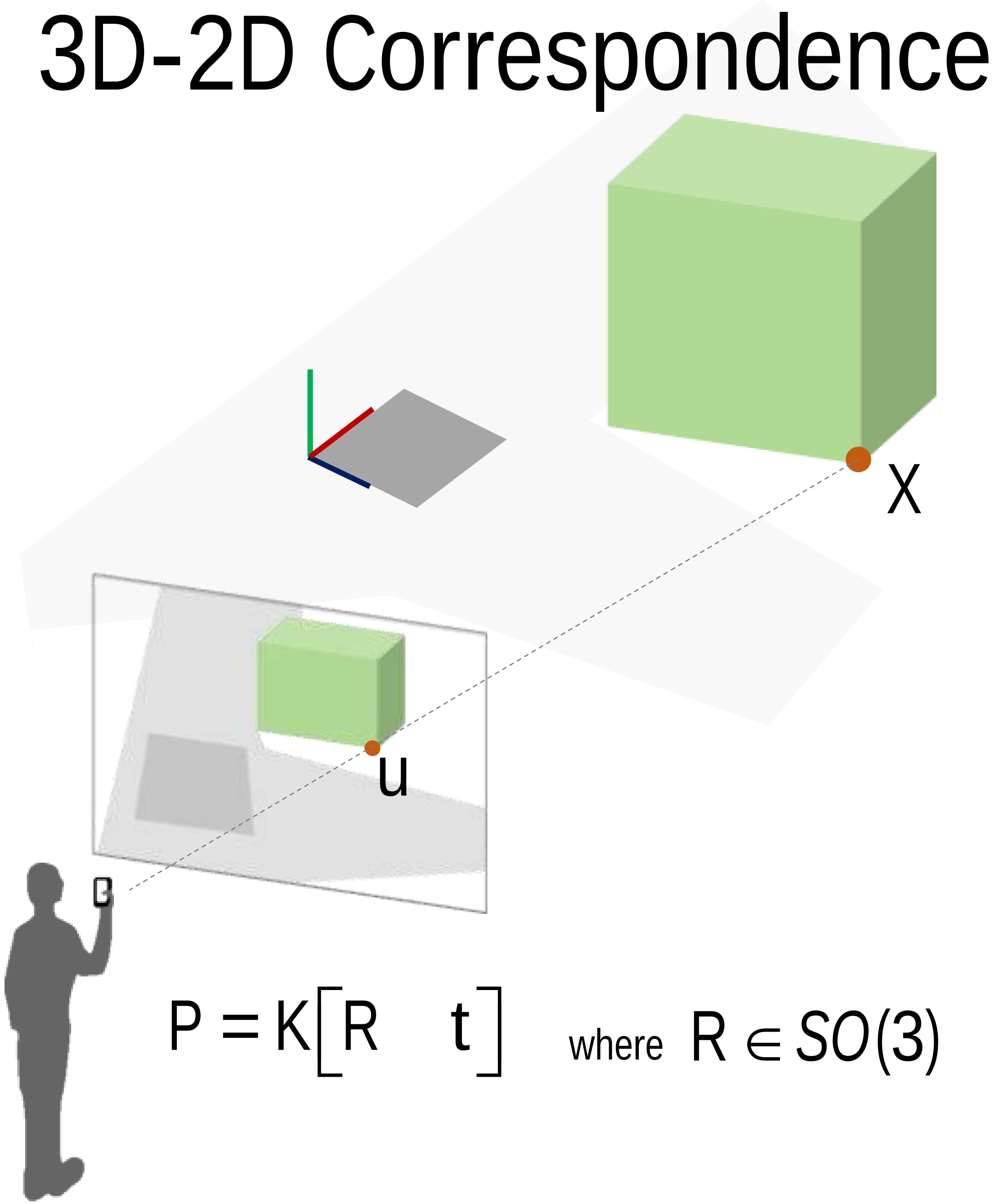
$$P = K\begin{bmatrix} R & t \end{bmatrix} \quad \text{where } R \in SO(3)$$

3D-2D correspondence: $u \leftrightarrow X$

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

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

3D-2D Correspondence



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

3D-2D correspondence: $u \leftrightarrow X$

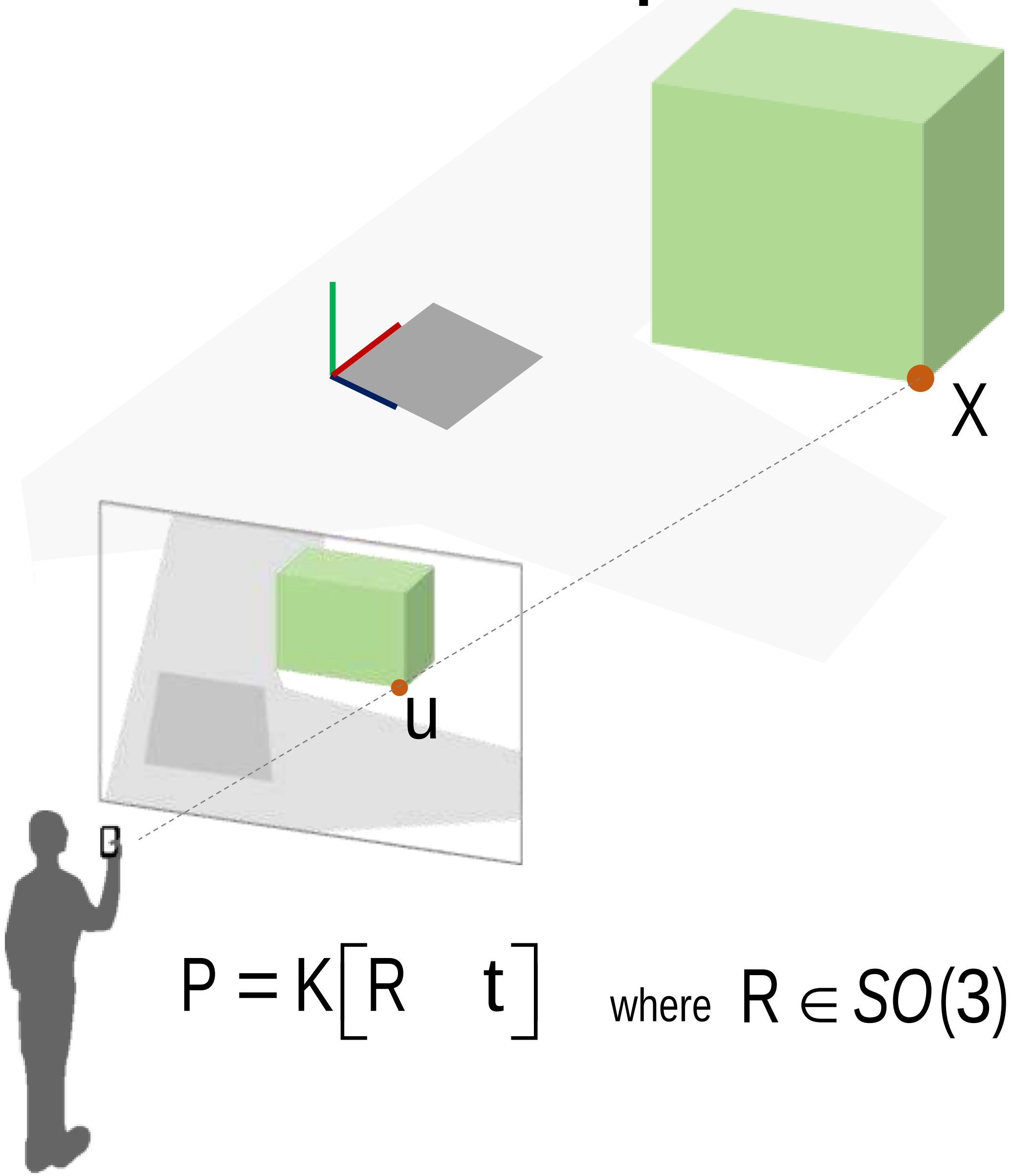
$$\lambda u = K\begin{bmatrix} R & t \end{bmatrix} X$$

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

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



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

3D-2D correspondence: $u \leftrightarrow X$

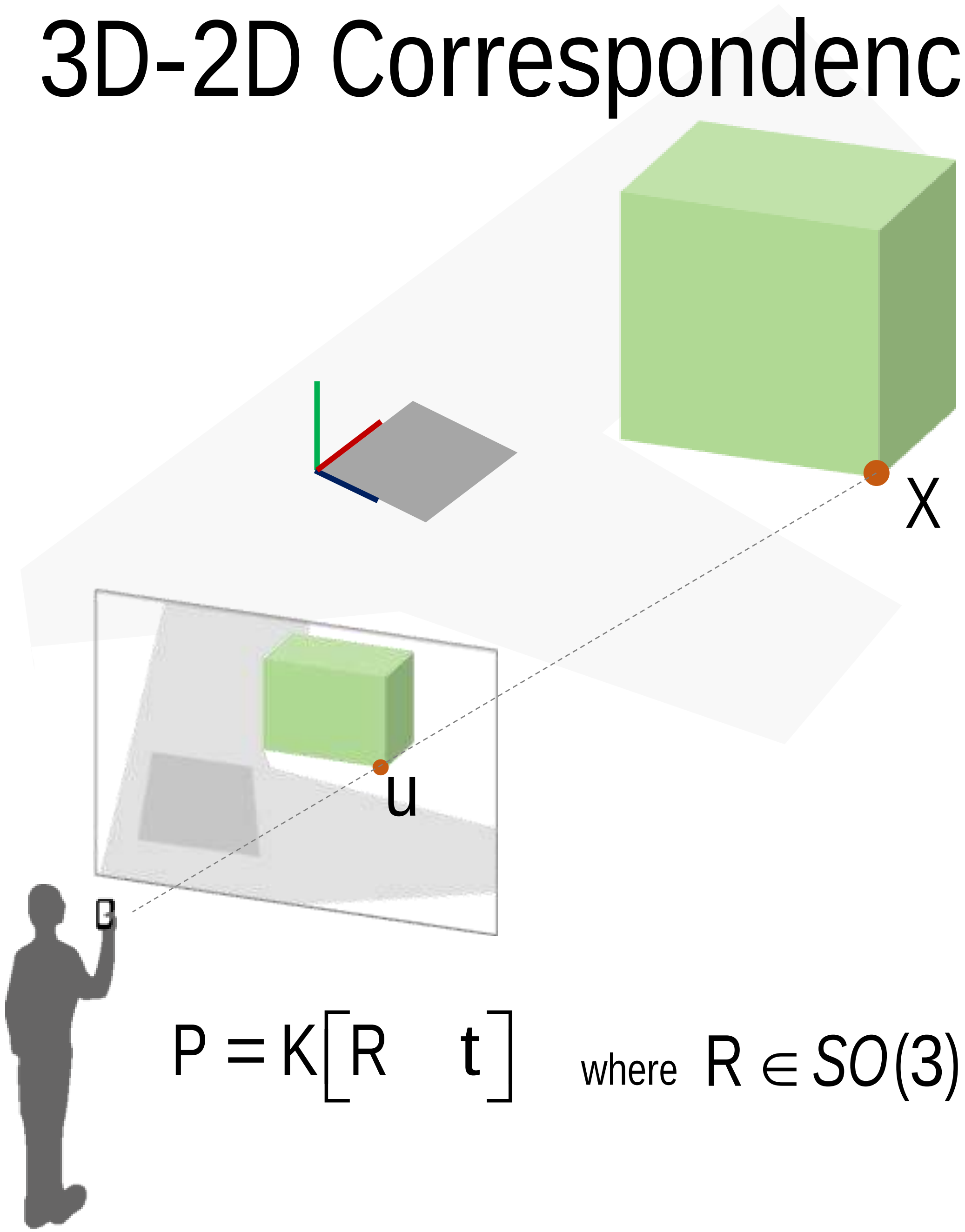
$$\lambda u = K\begin{bmatrix} R & t \end{bmatrix} X$$

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

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

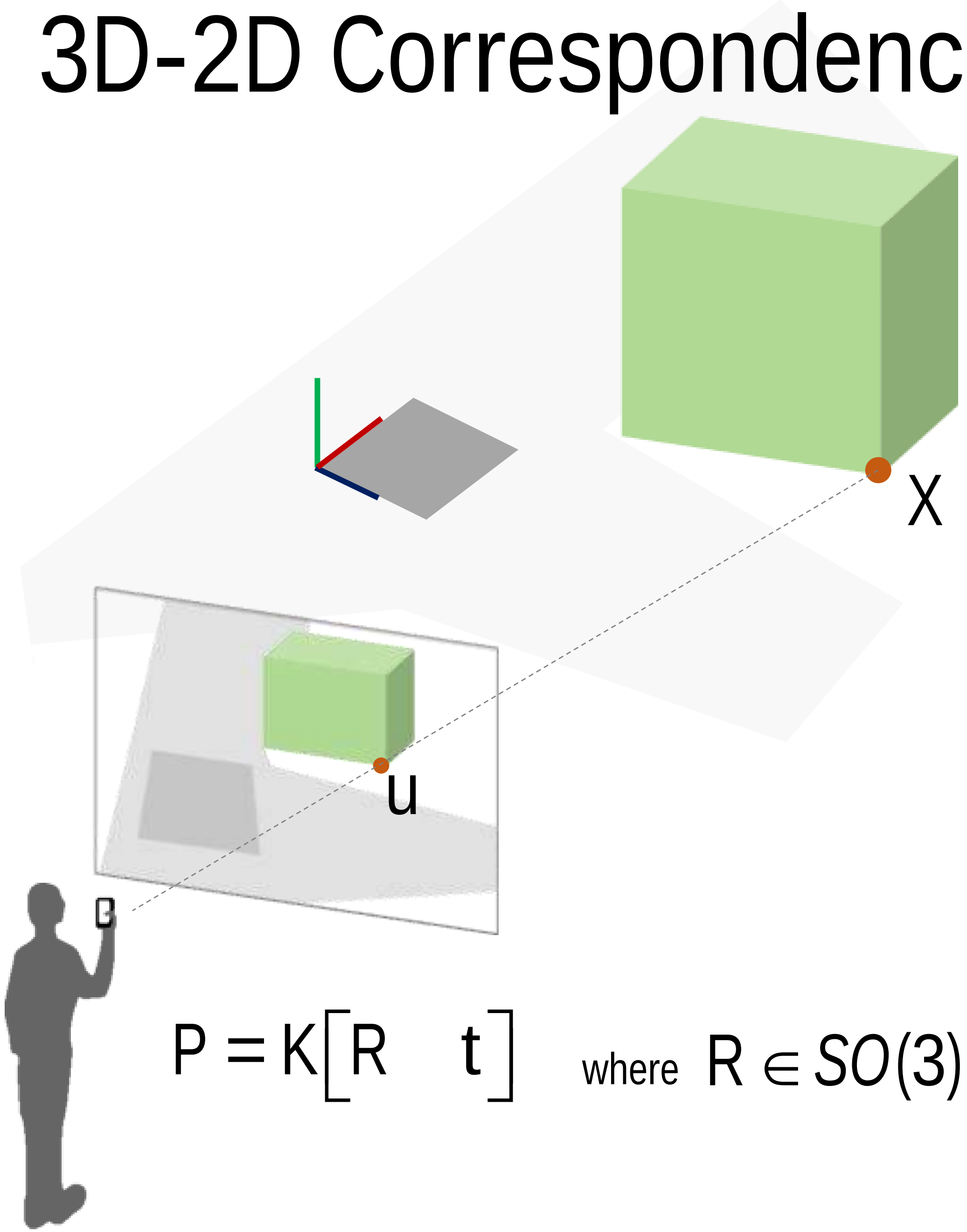


$$P = K\begin{bmatrix} R & t \end{bmatrix} \quad \text{where } R \in SO(3)$$

3D-2D correspondence: $u \leftrightarrow X$

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

3D-2D Correspondence



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

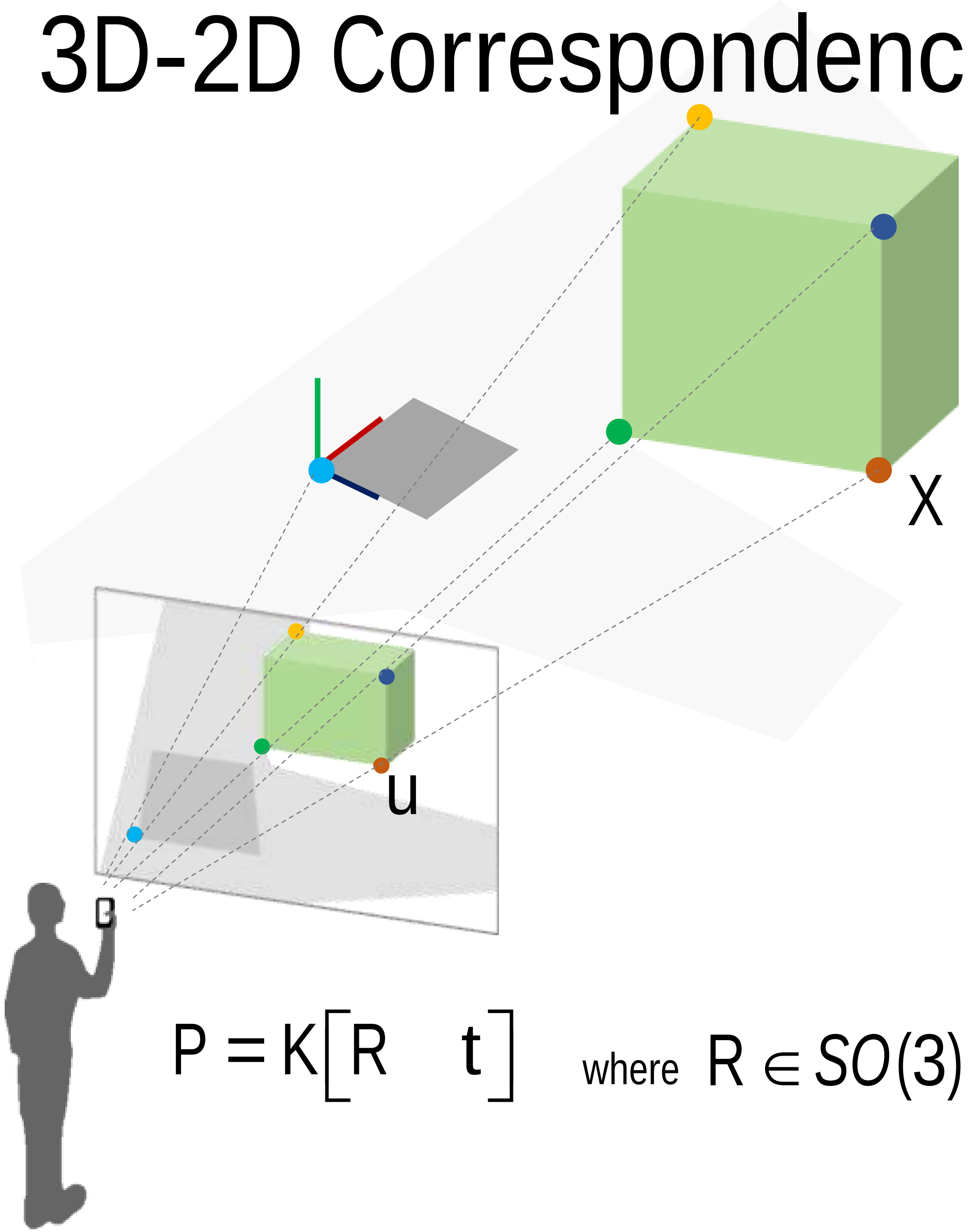
3D-2D correspondence: $u \leftrightarrow X$

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

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

2x12

3D-2D Correspondence



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

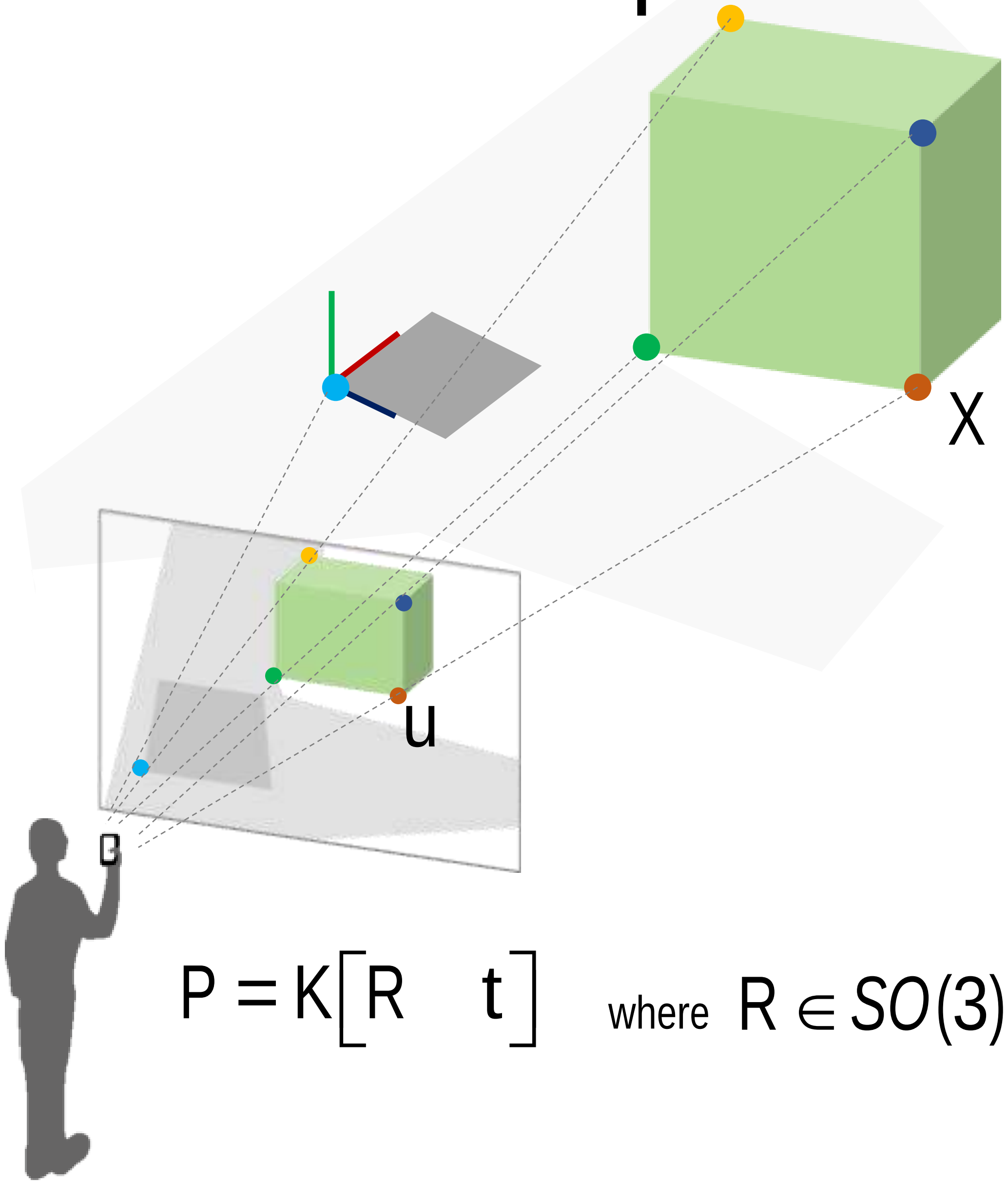
3D-2D correspondence: $u \leftrightarrow X$

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

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

$2m$x12

3D-2D Correspondence



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

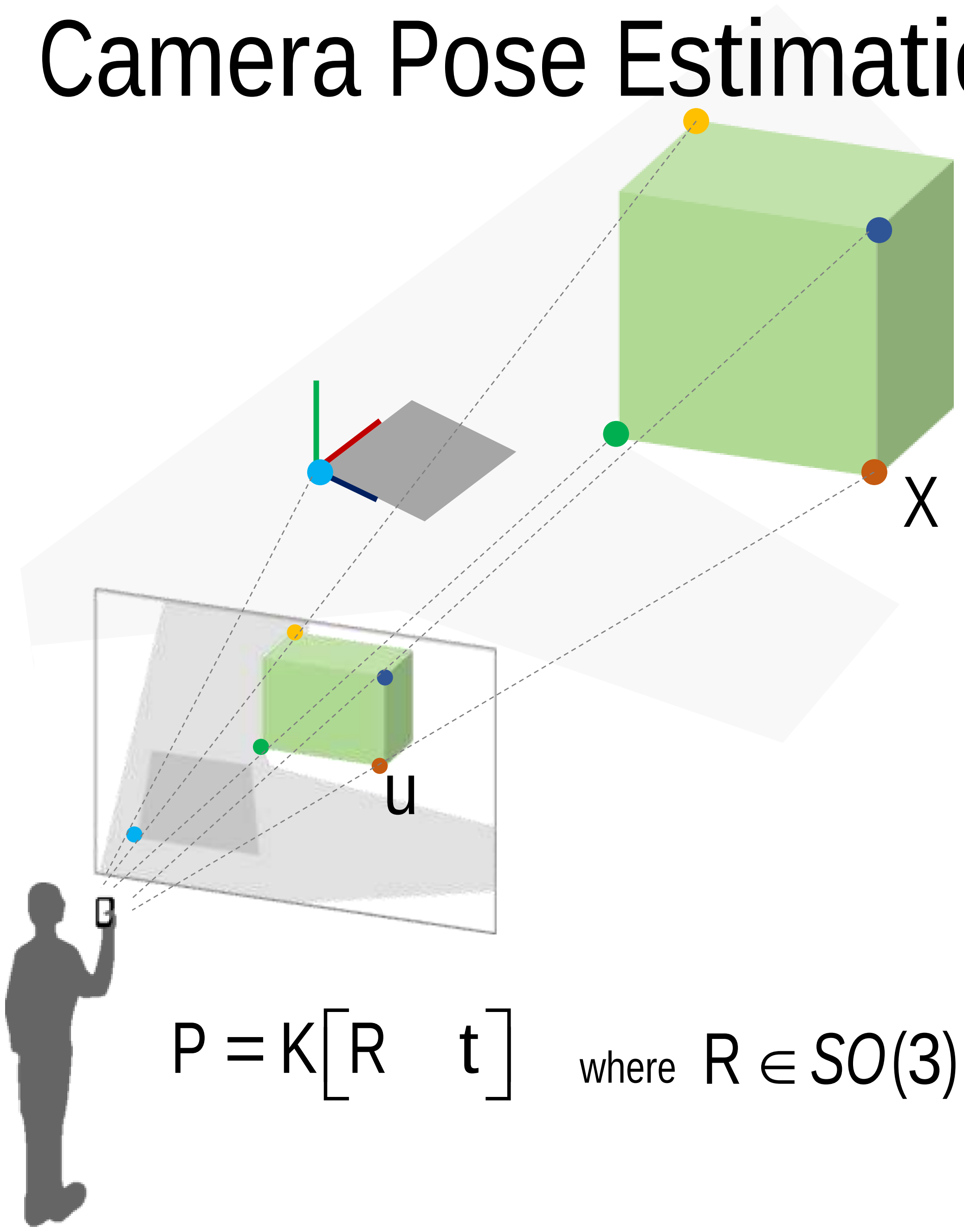
3D-2D correspondence: $u \leftrightarrow X$

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

$$\underbrace{\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & & & & & -u_1^xX & -u_1^xY & -u_1^xZ & -u_1^x \\ & & & & X_1 & Y_1 & Z_1 & 1 & -u_1^yX & -u_1^yY & -u_1^yZ & -u_1^y \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ X_m & Y_m & Z_m & 1 & & & & & -u_m^xX & -u_m^xY & -u_m^xZ & -u_m^x \\ & & & & X_m & Y_m & Z_m & 1 & -u_m^yX & -u_m^yY & -u_m^yZ & -u_m^y \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}}_{0}$$

$2m$x12

Camera Pose Estimation



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

If K is given,

$$K\begin{bmatrix} R & t \end{bmatrix} = \gamma\begin{bmatrix} p_1 & p_2 & p_3 & p_4 \end{bmatrix}$$

$$\longrightarrow \gamma R = K^{-1}\begin{bmatrix} p_1 & p_2 & p_3 \end{bmatrix}$$

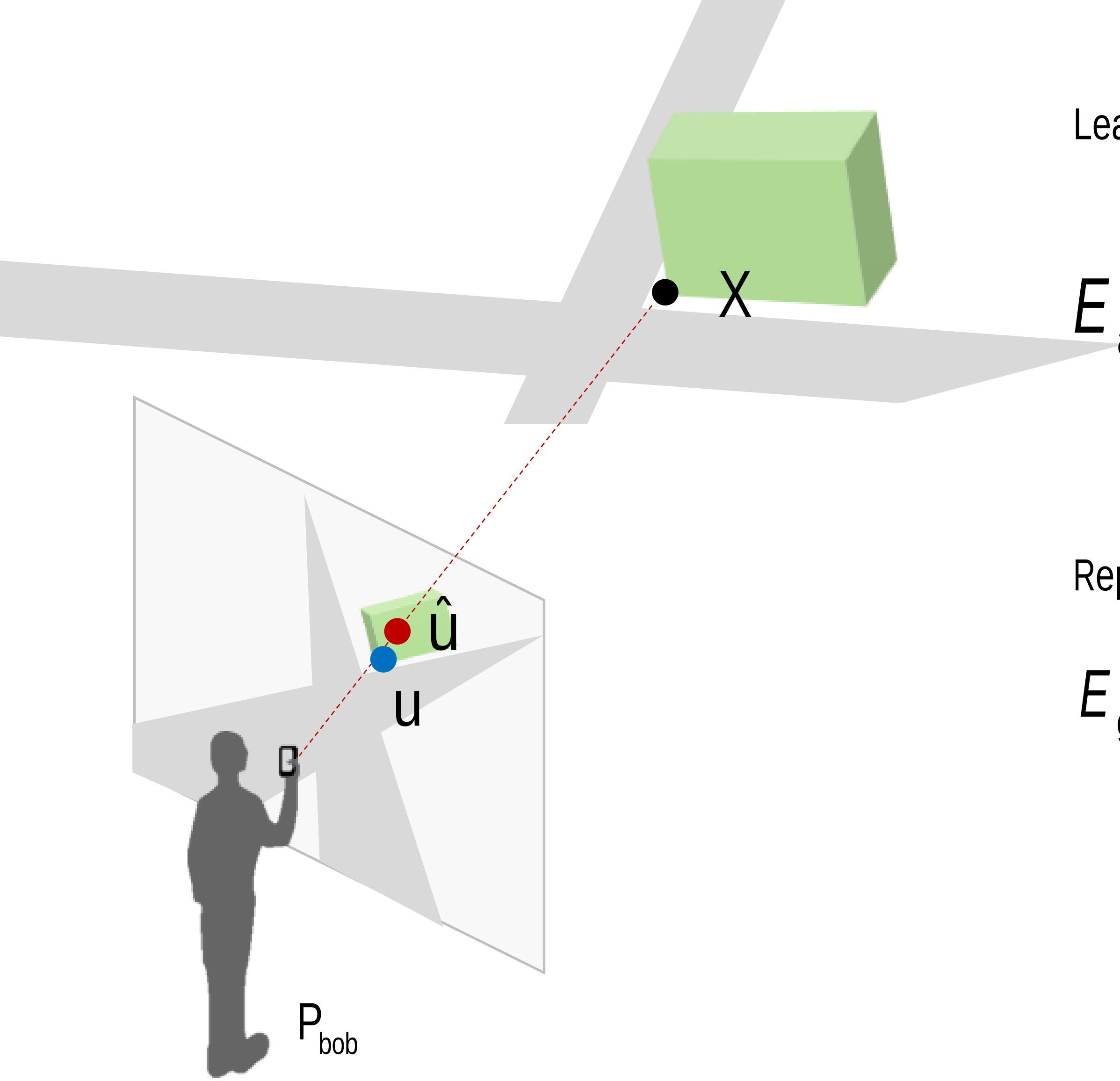
$$K^{-1}\begin{bmatrix} p_1 & p_2 & p_3 \end{bmatrix} = U\begin{bmatrix} d_{11} & & \\ & d_{22} & \\ & & d_{33} \end{bmatrix}V^T$$

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

$R = UV^T$: SVD cleanup

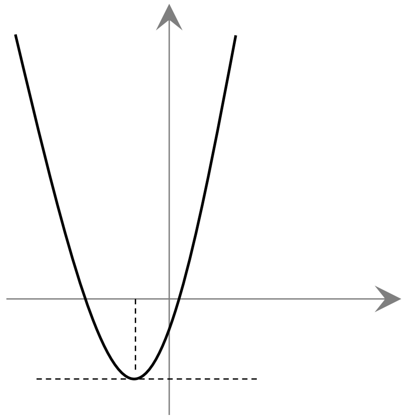
$$\longrightarrow t = \frac{K^{-1}p_4}{d_{11}}$$: Translation and scale recovery

Algebraic vs. Geometric error



Least squares solution (algebraic error):

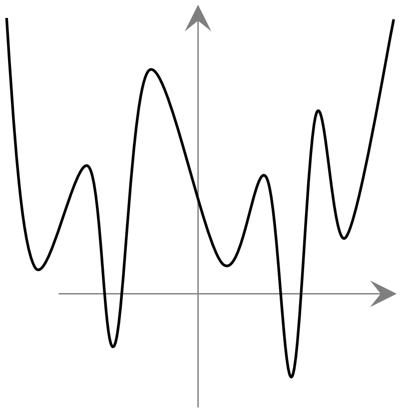
$$E_{alge} = \| A x - b \|^2$$



Reprojection error (geometric error):

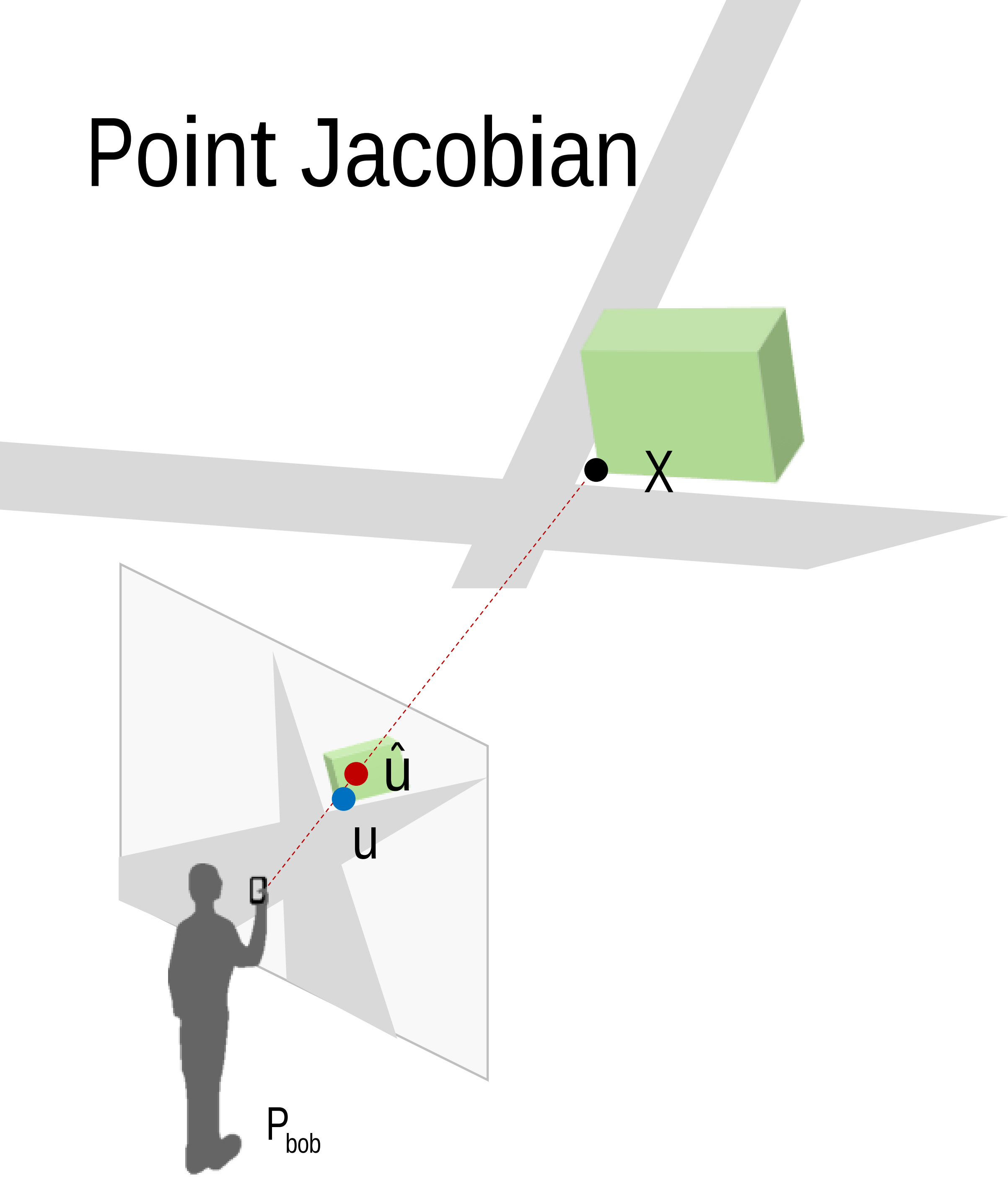
$$E_{geom} = \| \hat{u} - u \|^2$$

$$= \left(\frac{P_1 X}{P_3 X} - u_1 \right)^2 + \left(\frac{P_2 X}{P_3 X} - u_2 \right)^2$$



Point Jacobian

Black: given variables
Red: unknowns

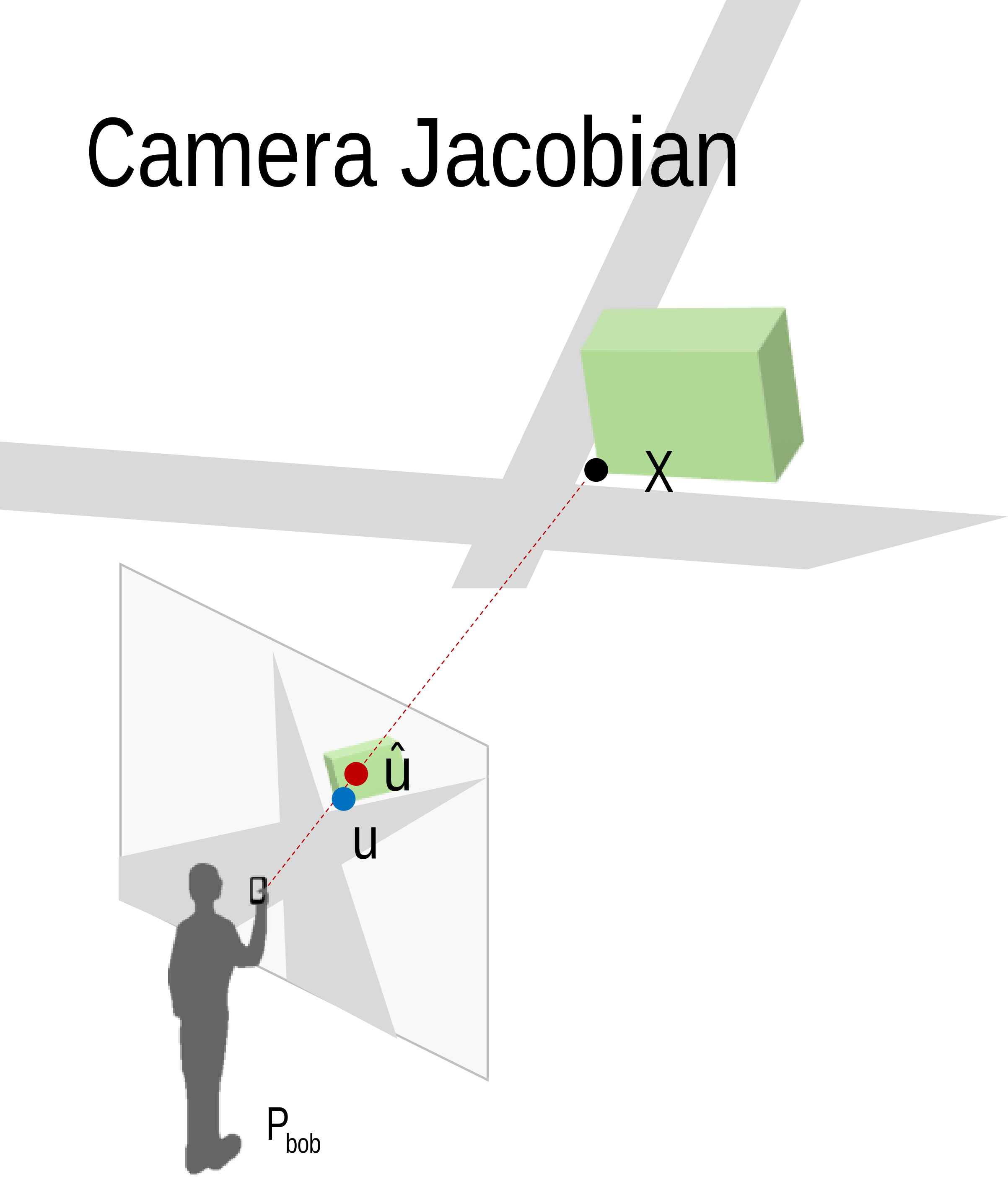


$$E_{geom} = \left\| \hat{u} - u \right\|^2$$

$$= \left(\frac{P_1 X}{P_3 X} - x \right)^2 + \left(\frac{P_2 X}{P_3 X} - y \right)^2$$

Camera Jacobian

Black: given variables
Red: unknowns

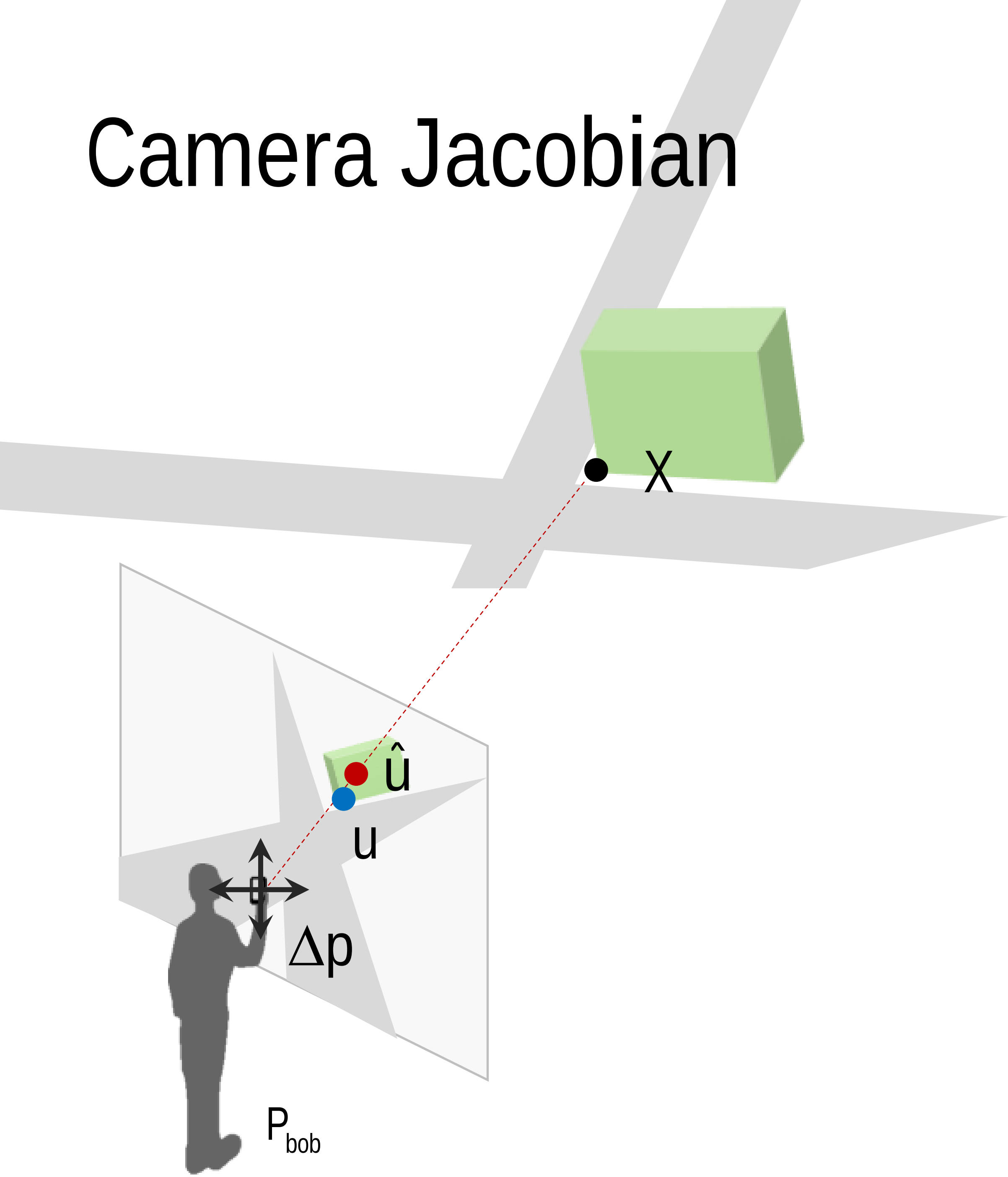


$$E_{geom} = \left\| \hat{\mathrm{u}} - \mathrm{u} \right\|^2$$

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

Black: given variables
Red: unknowns

Camera Jacobian



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

$$= \left(\frac{P_1 X}{P_3 X} - x \right)^2 + \left(\frac{P_2 X}{P_3 X} - y \right)^2$$

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

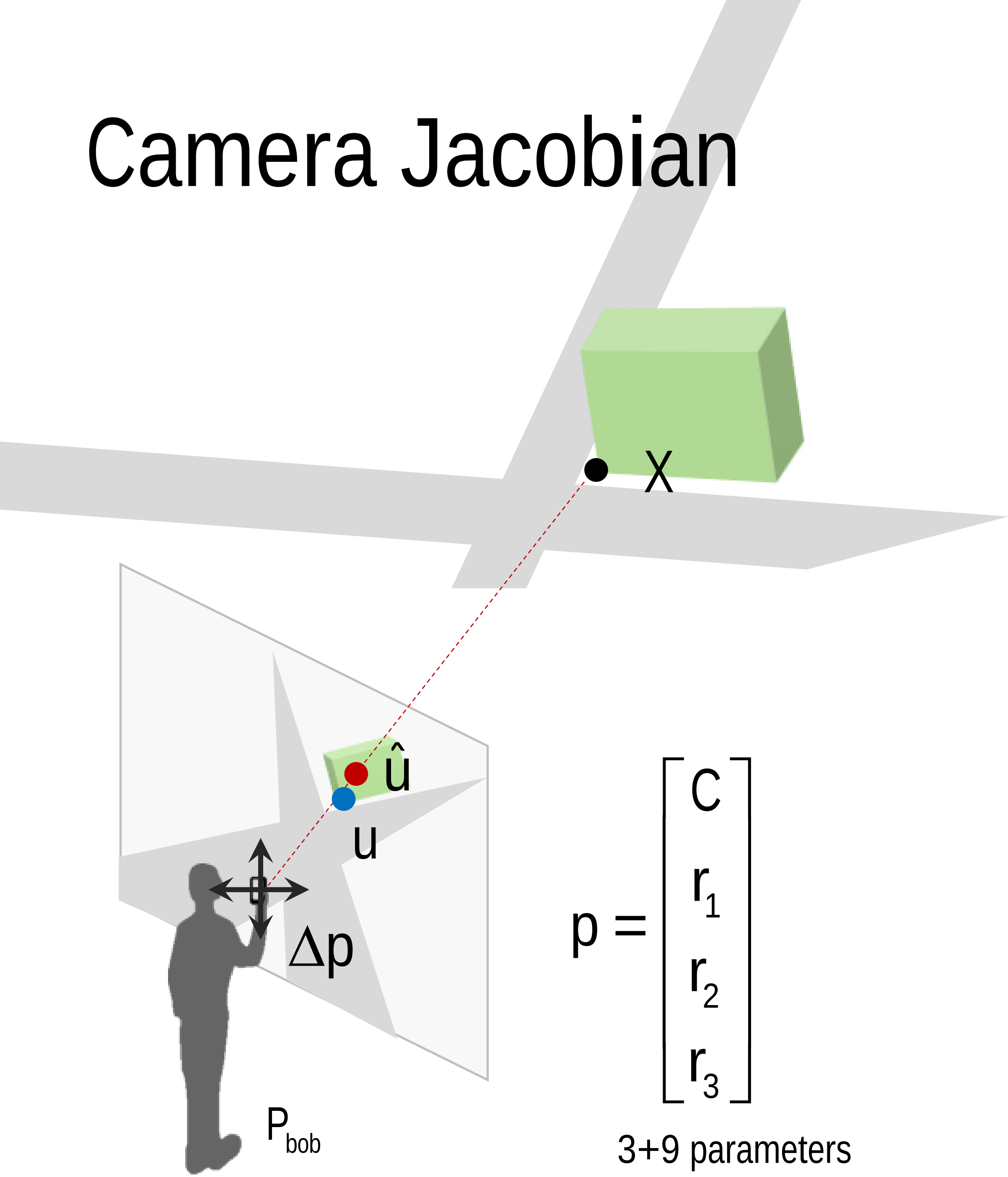
Camera Jacobian

Black: given variables
Red: unknowns

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

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

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

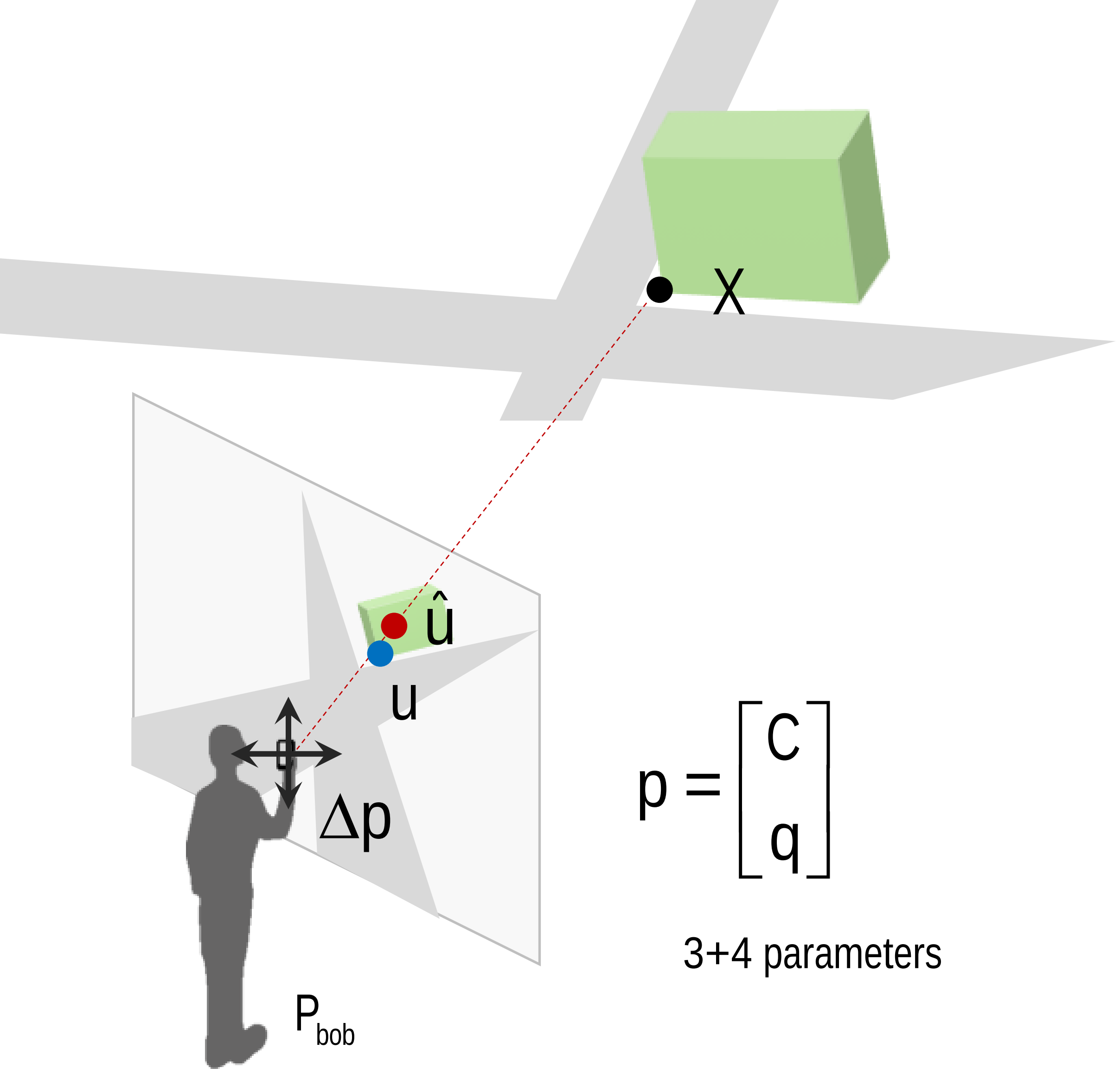


$$p = \begin{bmatrix} C \\ r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

3+9 parameters

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

Camera Jacobian



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

3+4 parameters

Black: given variables

Red: unknowns

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

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

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

$$\rightarrow \quad \frac{\partial}{\partial q}\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \frac{\partial}{\partial R}\begin{bmatrix} u \\ v \\ w \end{bmatrix}\frac{\partial R}{\partial q}$$

: Chain rule

Quaternion jacobian

Quaternion Jacobian

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

where

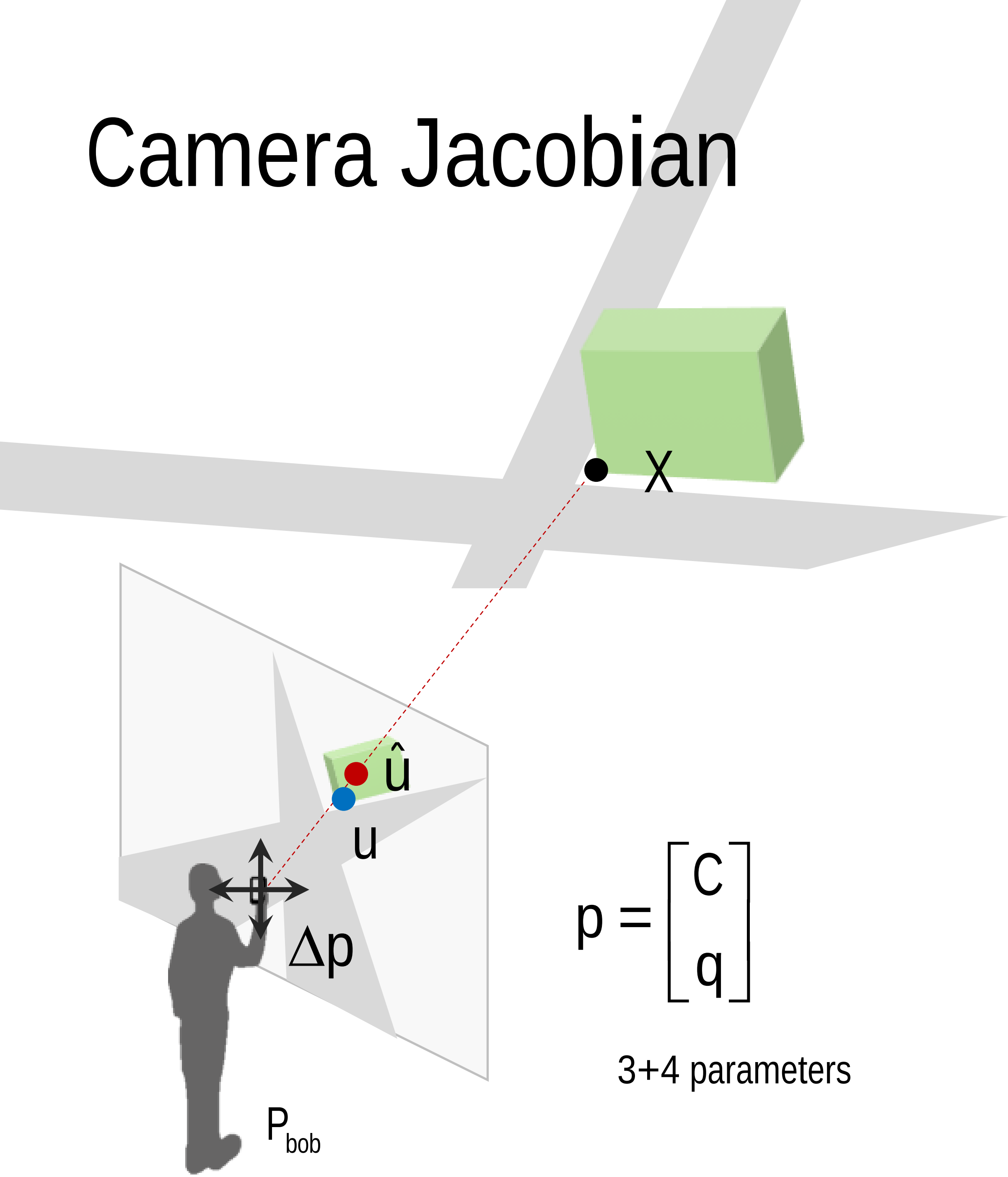
$$q = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix}^T$$

$$\underset{9\text{x}4}{\frac{\partial R}{\partial q}} = \begin{bmatrix} 0 & 0 & -4q_y & -4q_z \\ -2q_z & 2q_y & 2q_x & -2q_w \\ 2q_y & 2q_z & 2q_w & 2q_x \\ 2q_z & 2q_y & 2q_x & 2q_w \\ 0 & -4q_x & 0 & -4q_z \\ -2q_x & -2q_w & 2q_z & 2q_y \\ -2q_y & 2q_z & -2q_w & 2q_x \\ 2q_x & 2q_w & 2q_z & 2q_y \\ 0 & -4q_x & -4q_y & 0 \end{bmatrix}$$

Camera Jacobian

Black: given variables
Red: unknowns

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



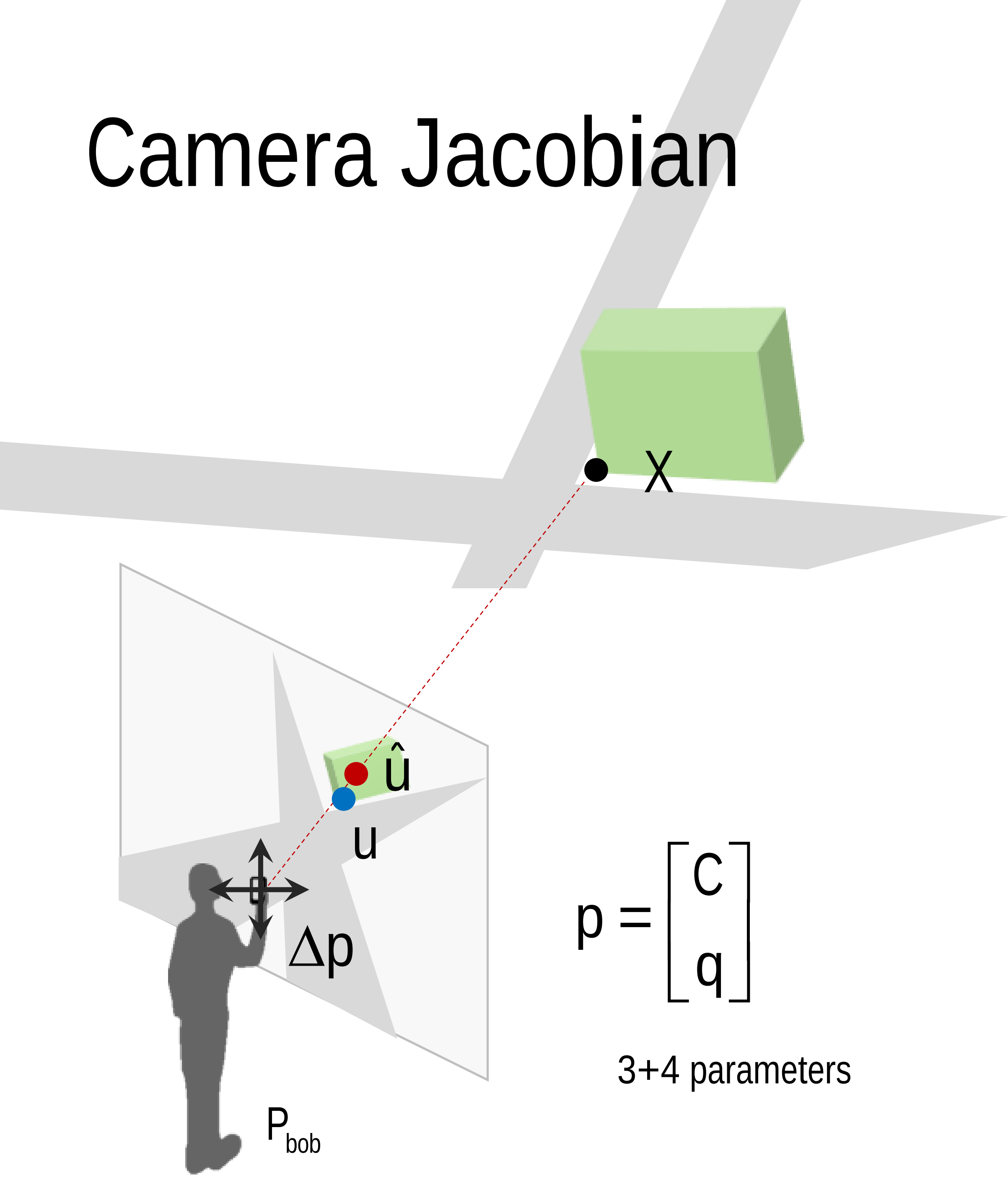
$$\mathrm{p} = \begin{bmatrix} \mathrm{C} \\ \mathrm{q} \end{bmatrix}$$

3+4 parameters

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

Camera Jacobian

Black: given variables
Red: unknowns



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

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

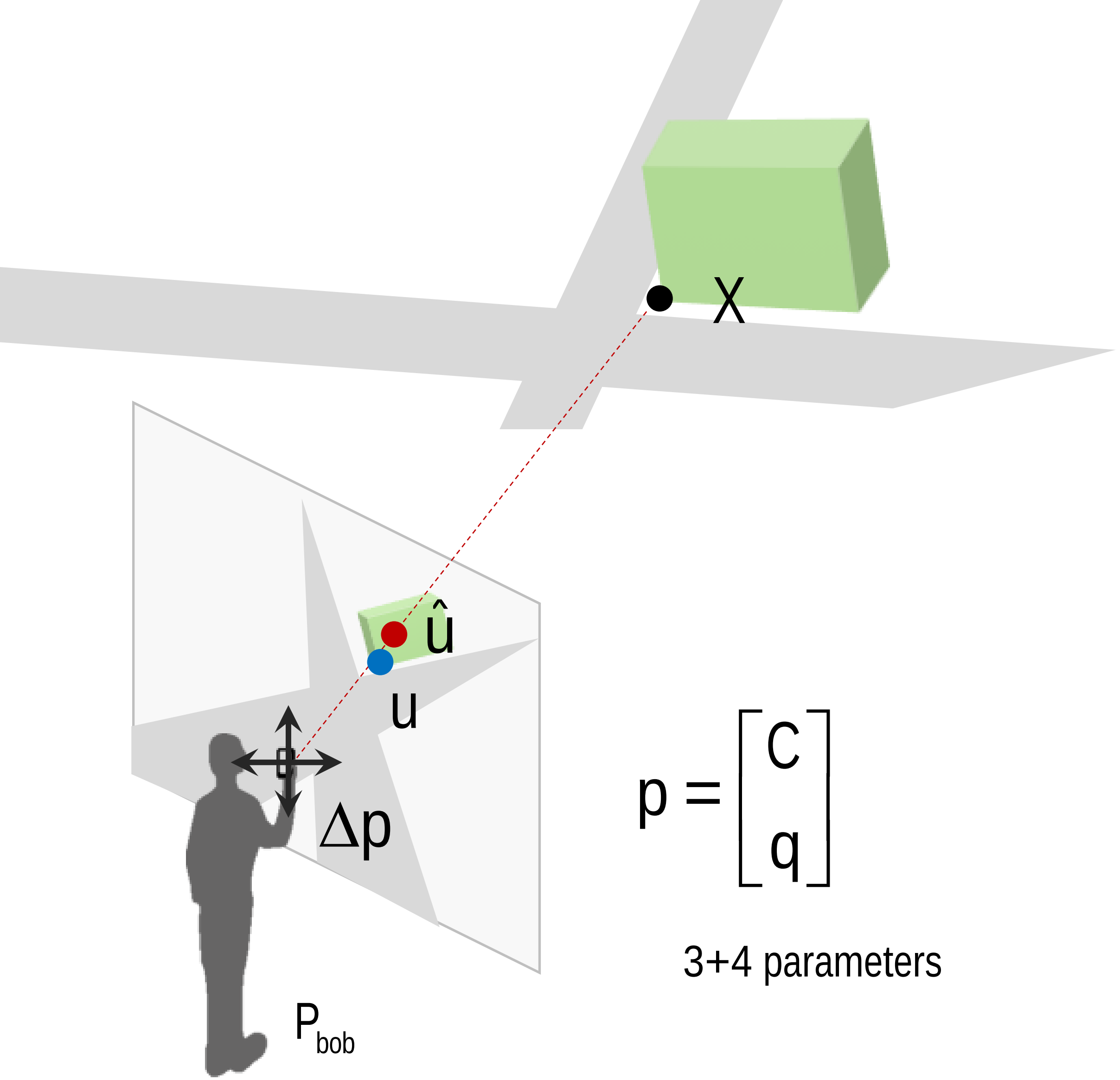
$$p = \begin{bmatrix} C \\ q \end{bmatrix}$$

3+4 parameters

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

Camera Jacobian

Black: given variables
Red: unknowns



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

3+4 parameters

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

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

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

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

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).

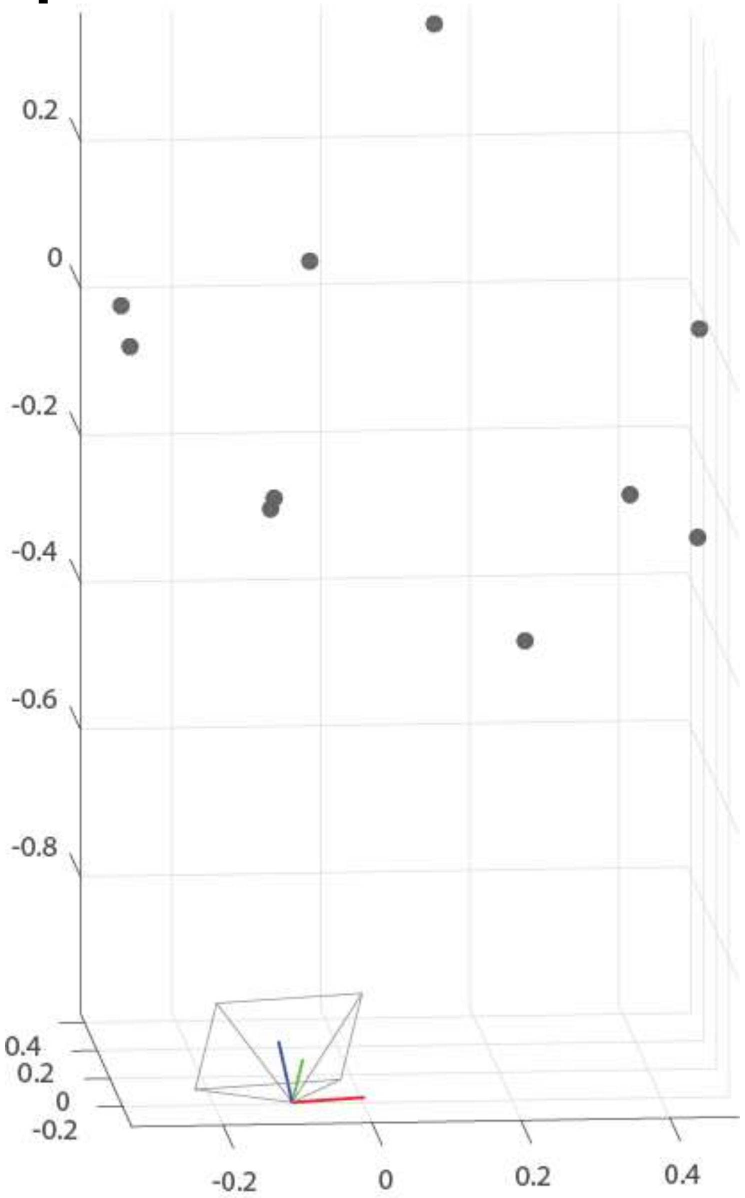
  $\mathbf{p} = \mathbf{p} + \Delta\mathbf{p}$

  Normalize the quaternion scale, $\mathbf{p}_{4:7} = \mathbf{p}_{4:7} / \|\mathbf{p}_{4:7}\|$.

end for

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

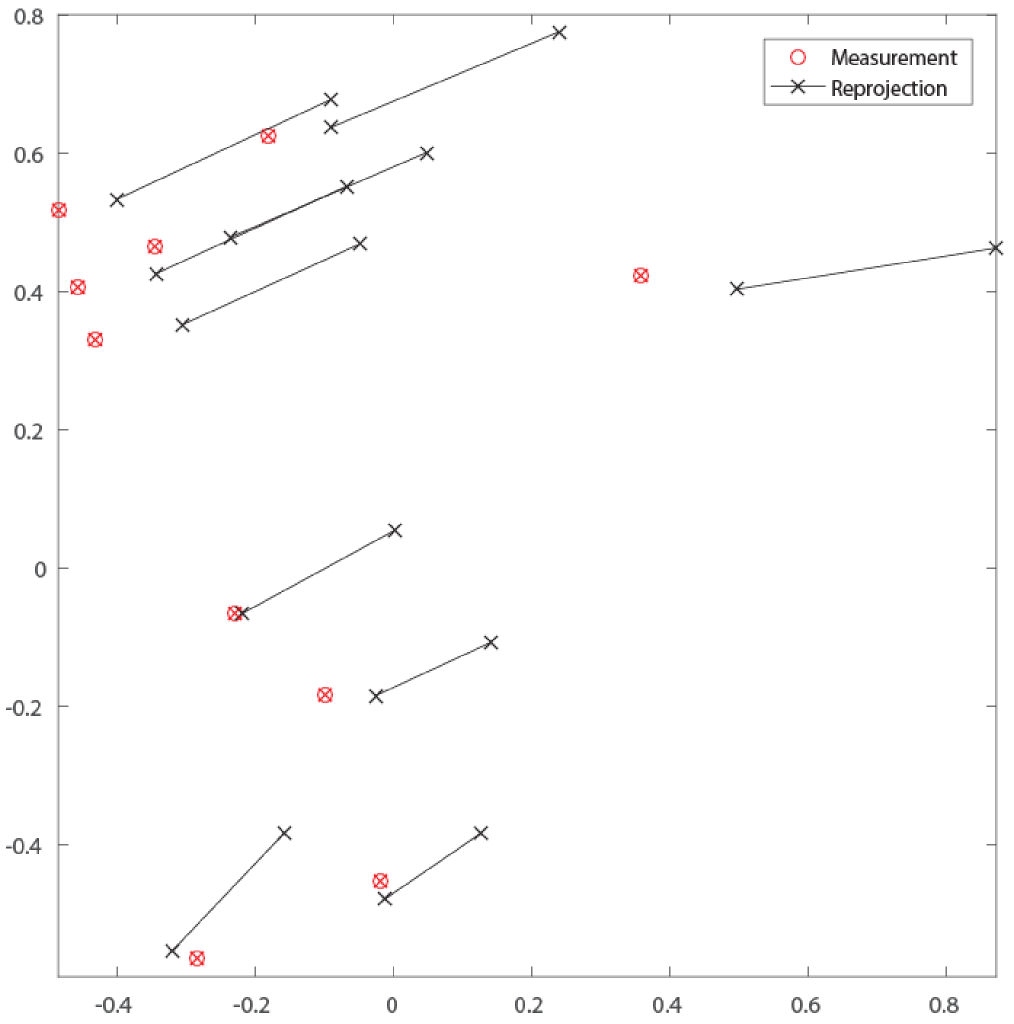
Example




```
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
```

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

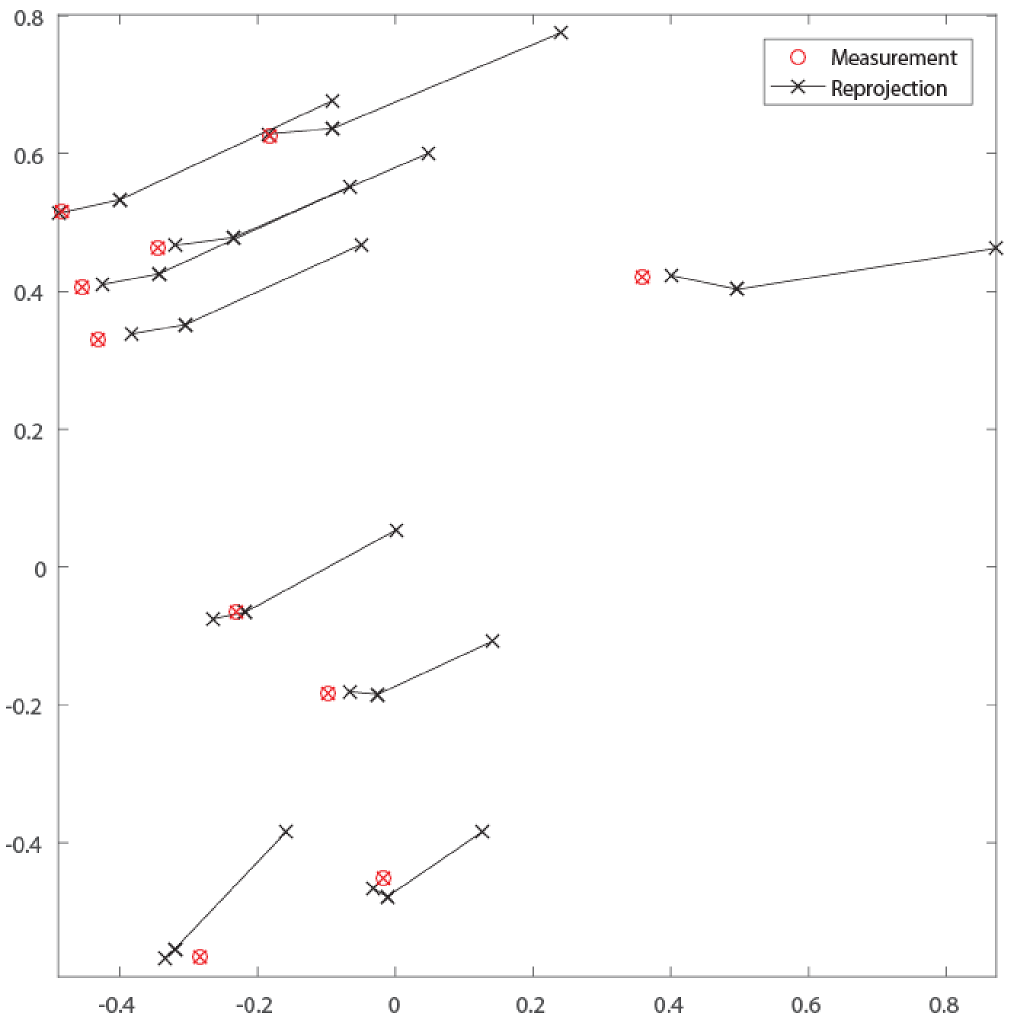
Example




```
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
```

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

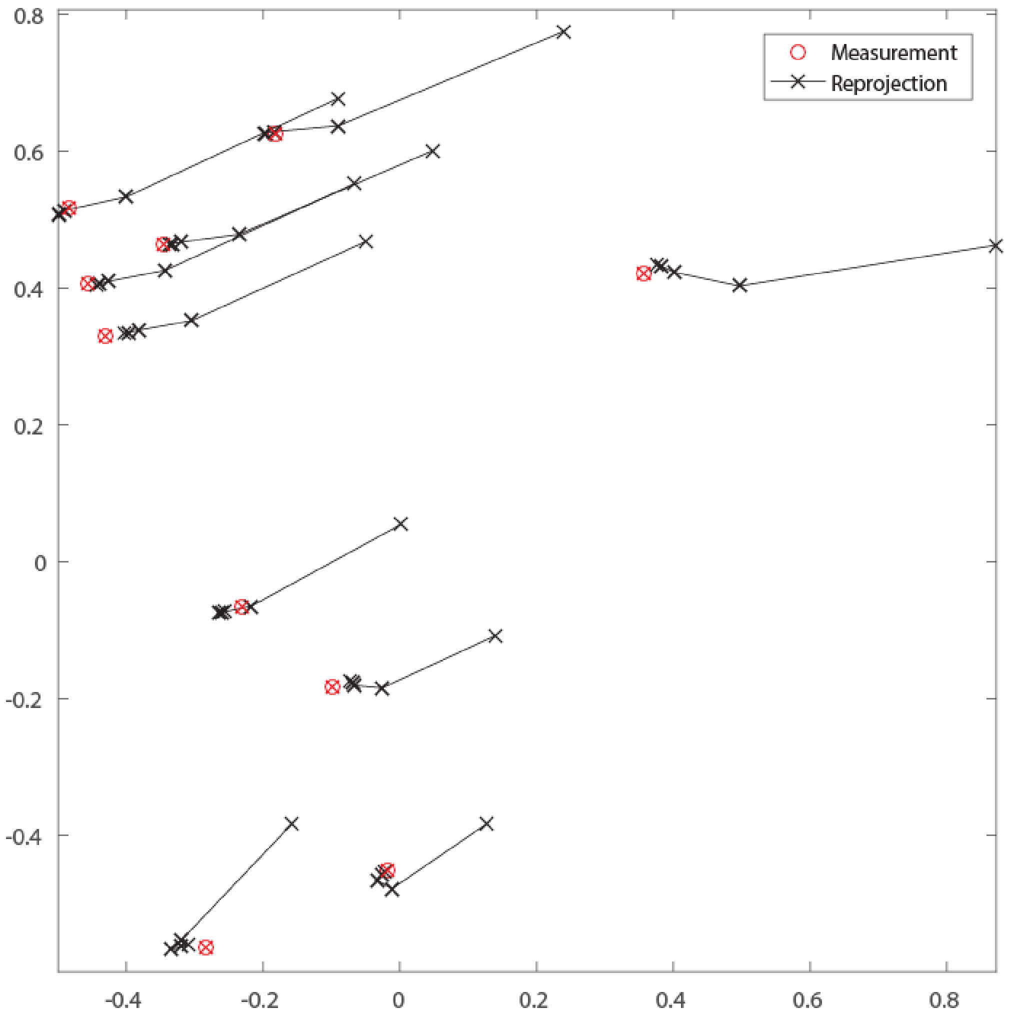
Example




```
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
```

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

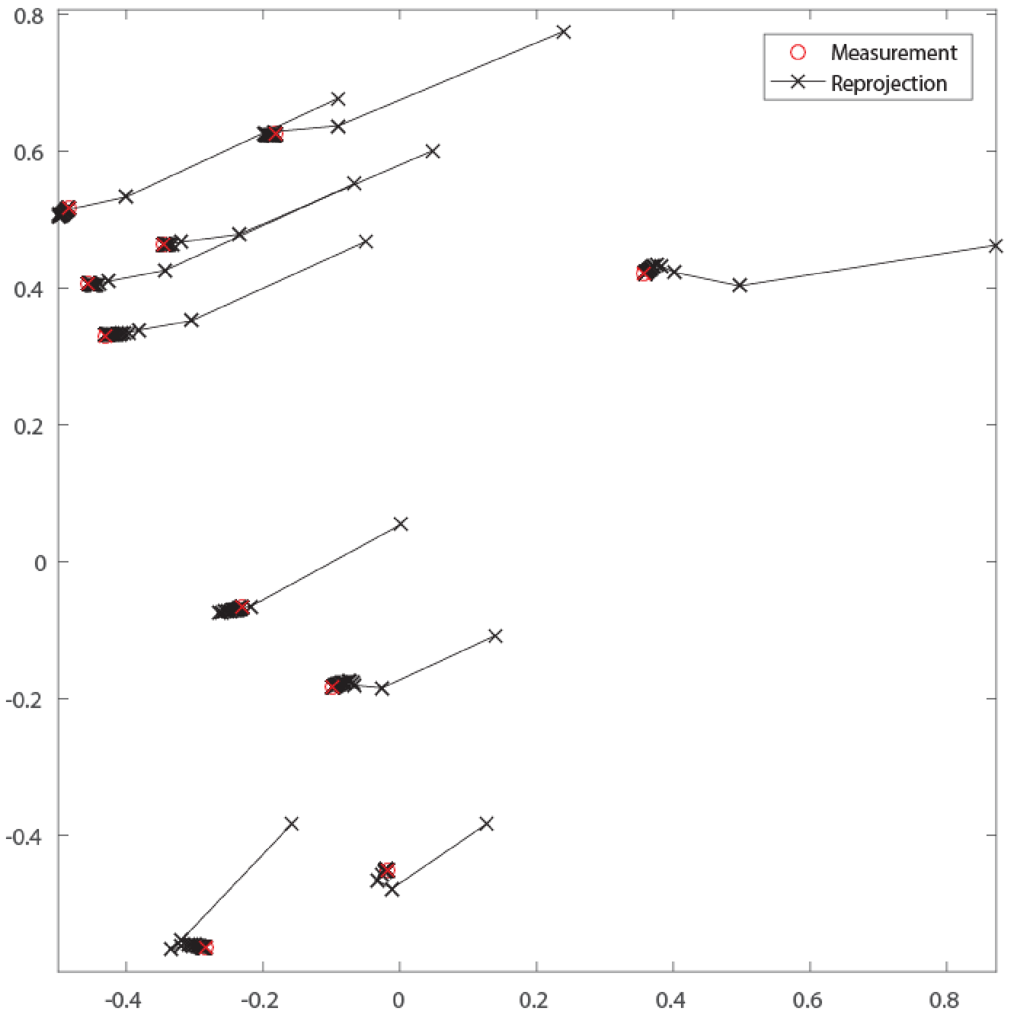
Example




```
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
```

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

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
```

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

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