

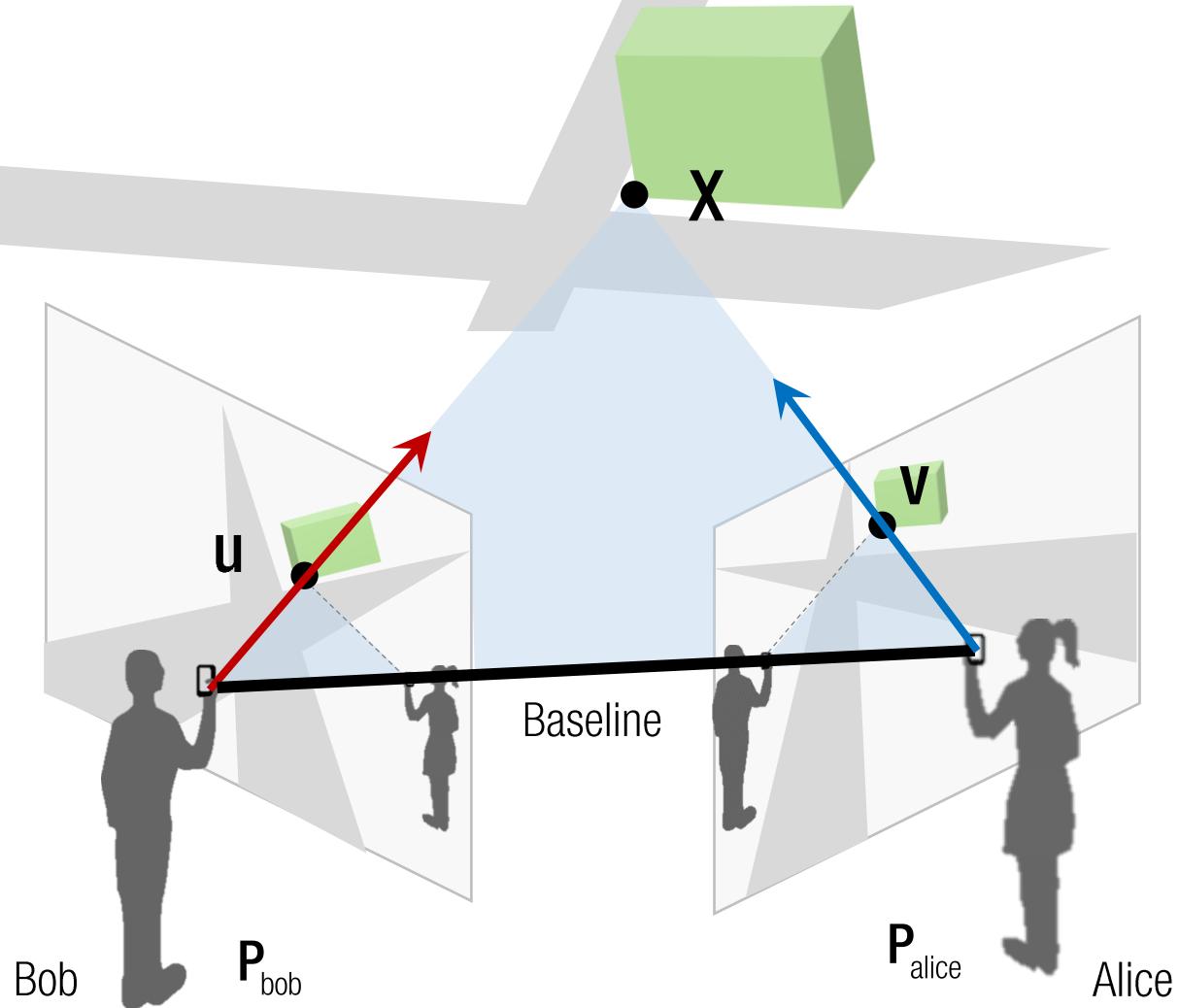


Nonlinear Point and Pose Refinement

Hyun Soo Park

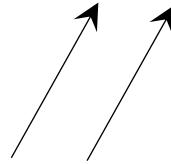
Triangulation Refinement

Recall: Triangulation



General camera pose

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



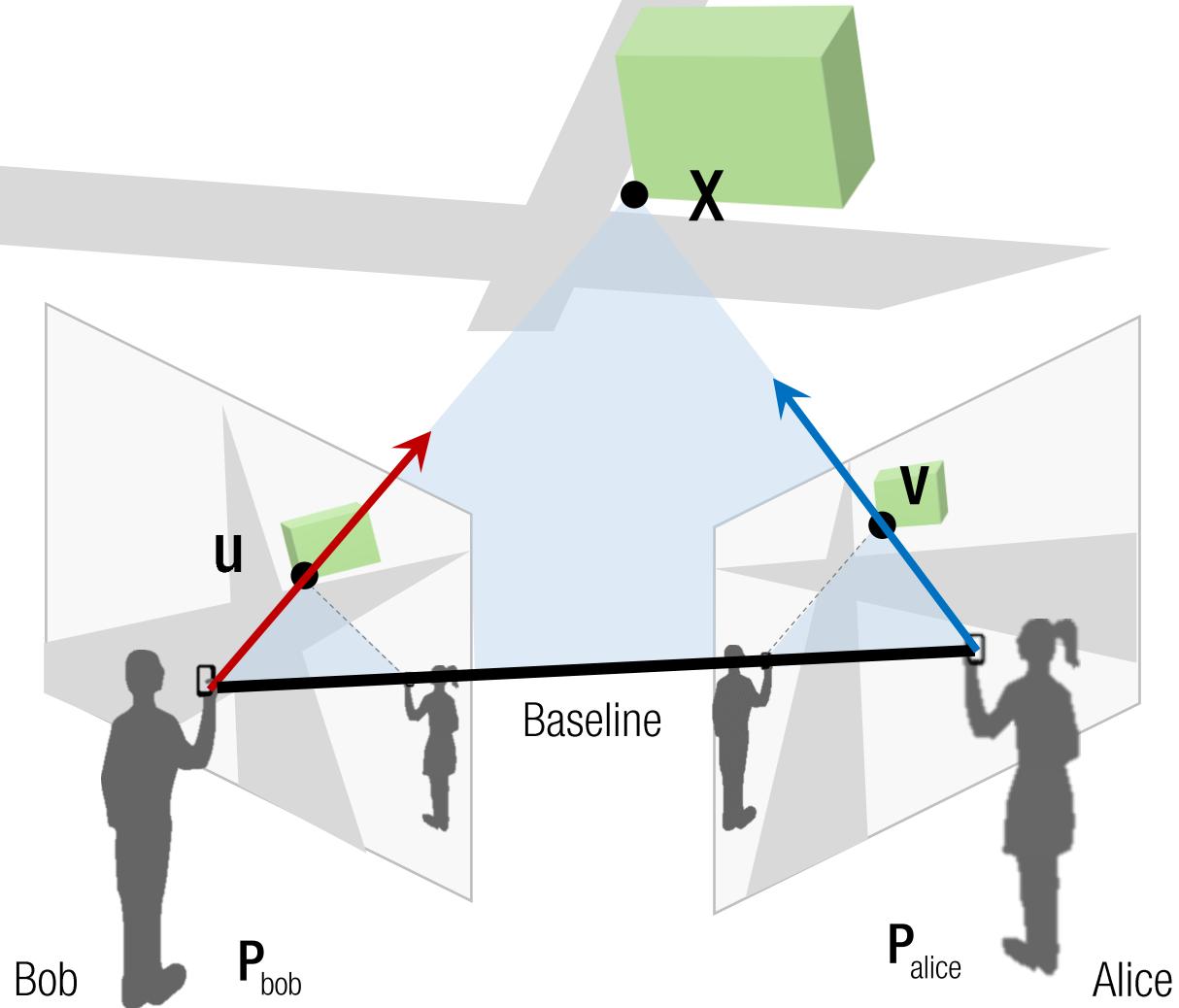
Two 3D vectors are parallel.

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

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

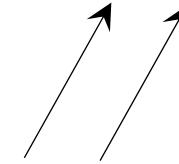
Skew-symmetric matrix

Recall: Triangulation



General camera pose

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



Two 3D vectors are parallel.

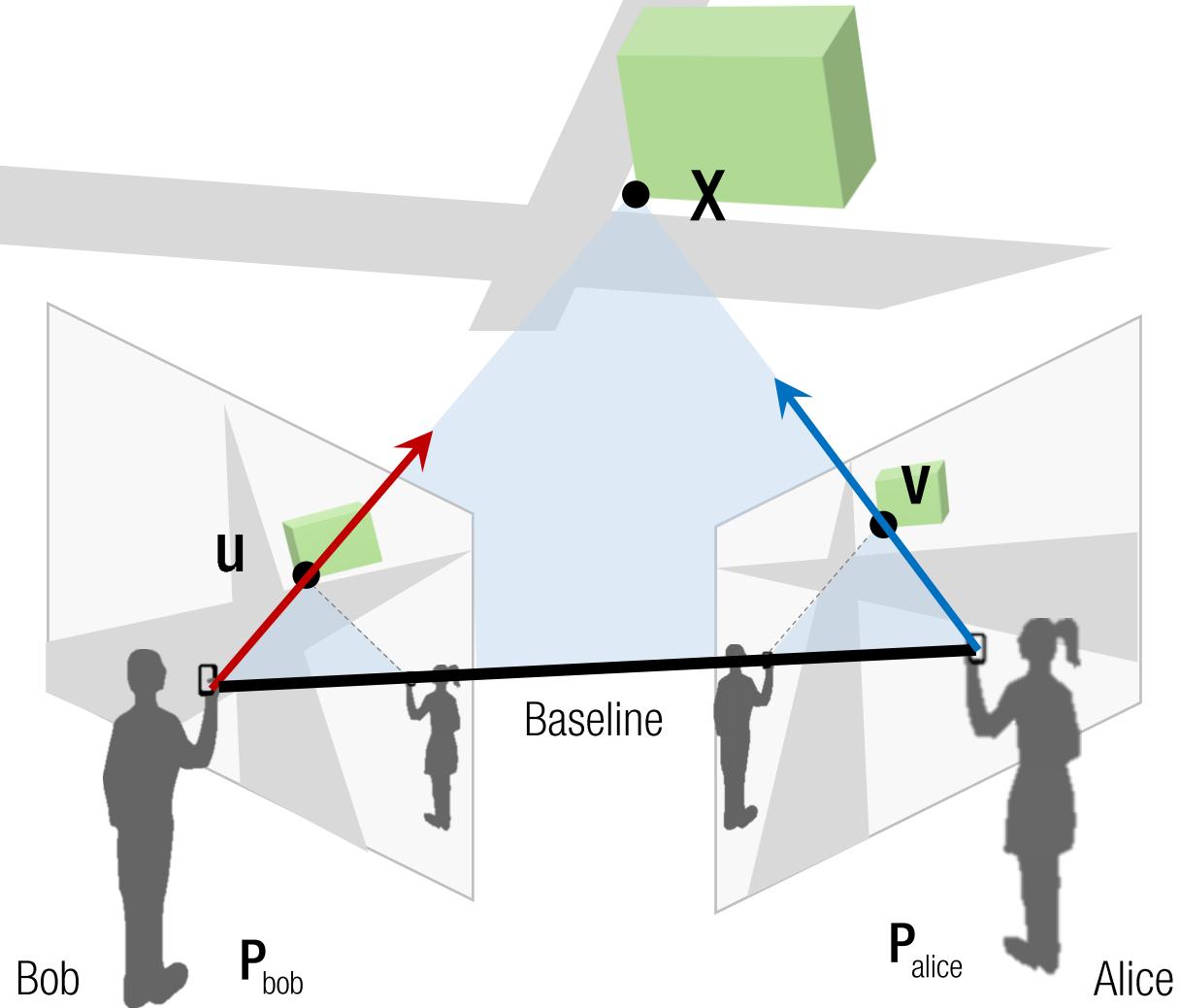
$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix} \times P_{\text{bob}} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

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

Knowns
Unknowns

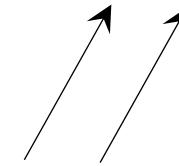
Skew-symmetric matrix

Recall: Triangulation



General camera pose

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



Two 3D vectors are parallel.

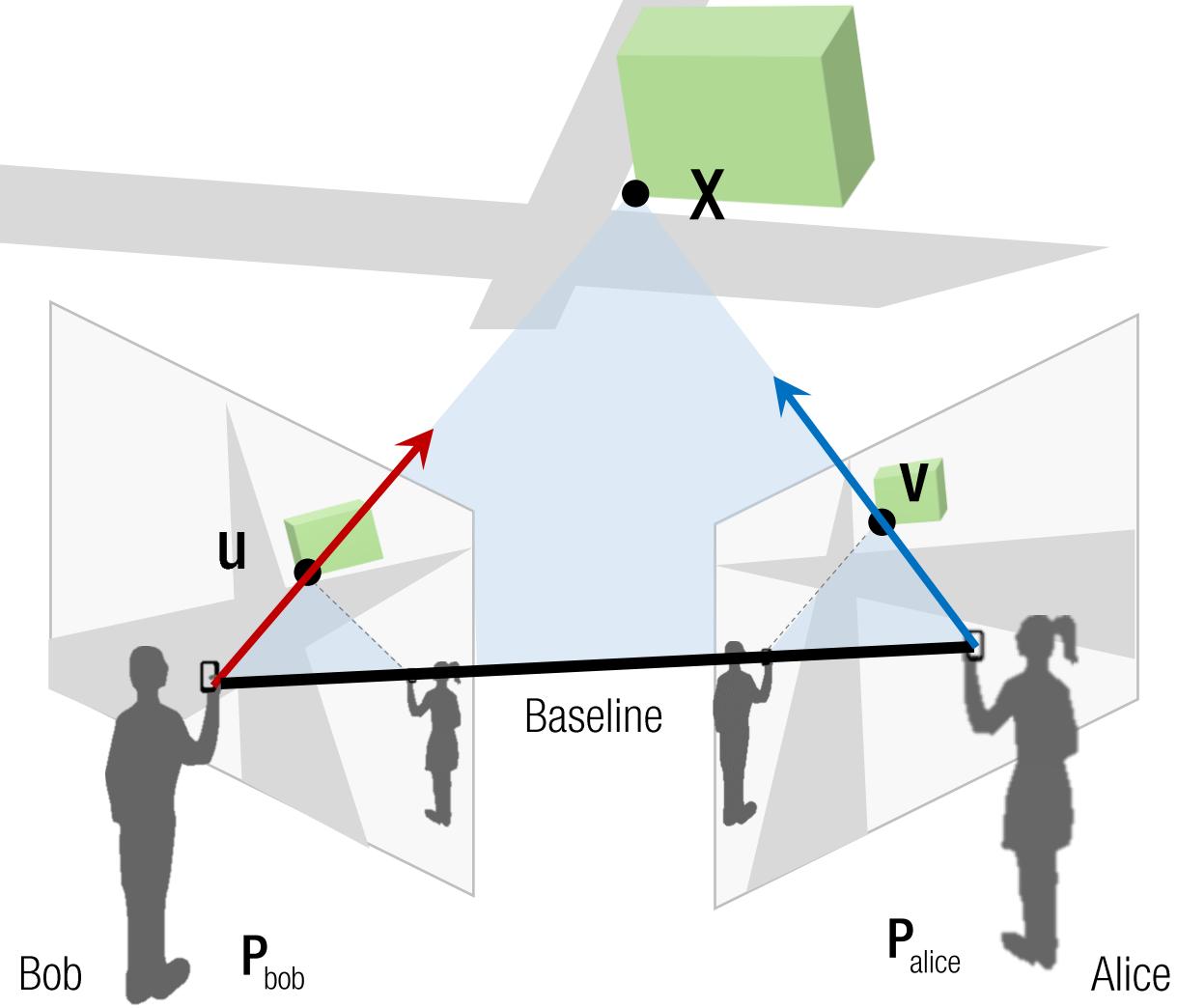
$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix} \times P_{\text{bob}} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

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

2x4

Knowns
Unknowns

Recall: Triangulation



General camera pose

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

Two 3D vectors are parallel.

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

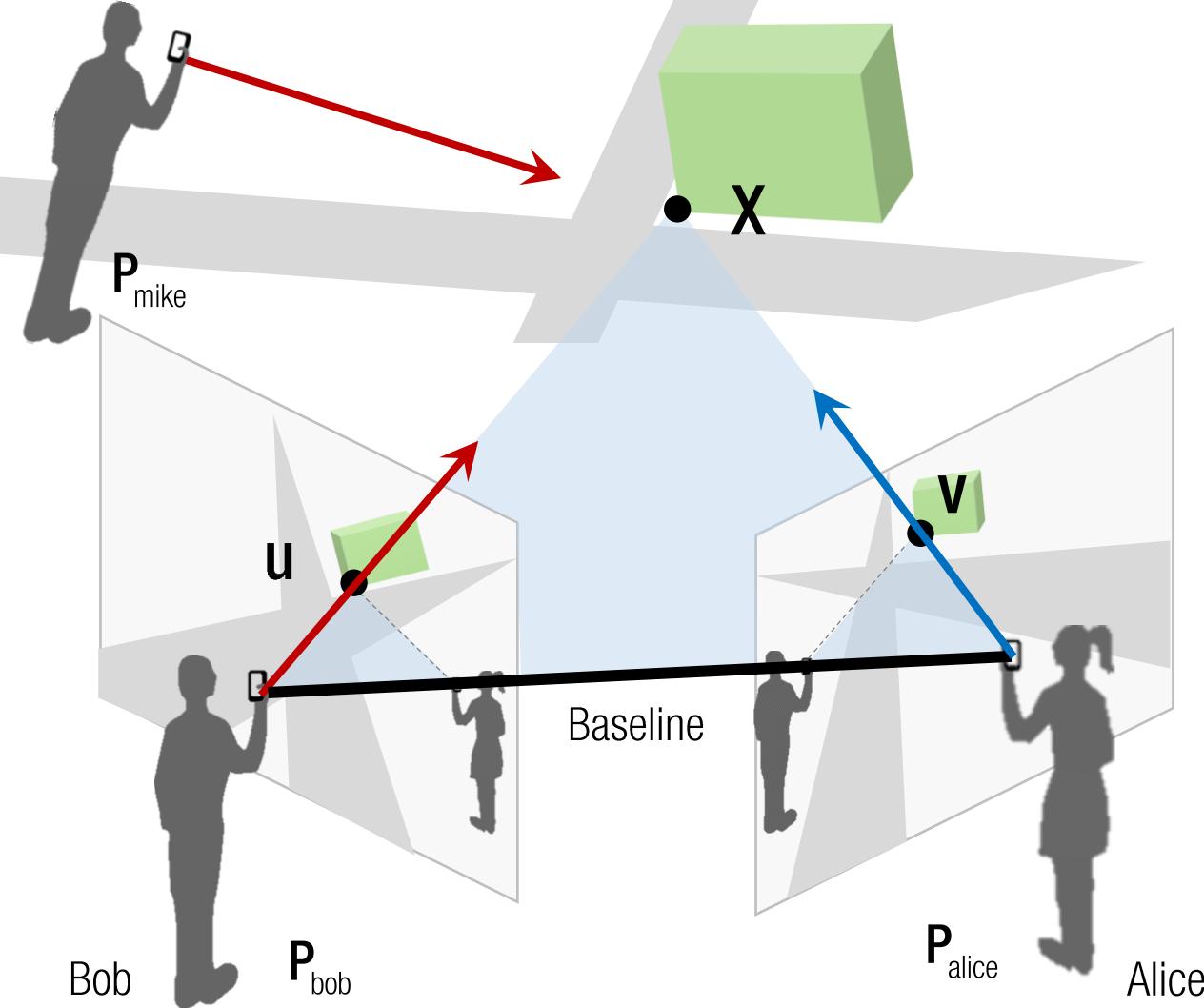
$$\rightarrow \begin{bmatrix} u \\ 1 \end{bmatrix} \times P_{\text{bob}} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

$$\begin{bmatrix} v \\ 1 \end{bmatrix} \times P_{\text{alice}} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

4x4

Knowns
Unknowns

Recall: Triangulation



General camera pose

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

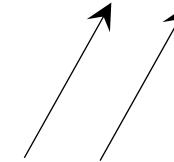
Two 3D vectors are parallel.

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

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

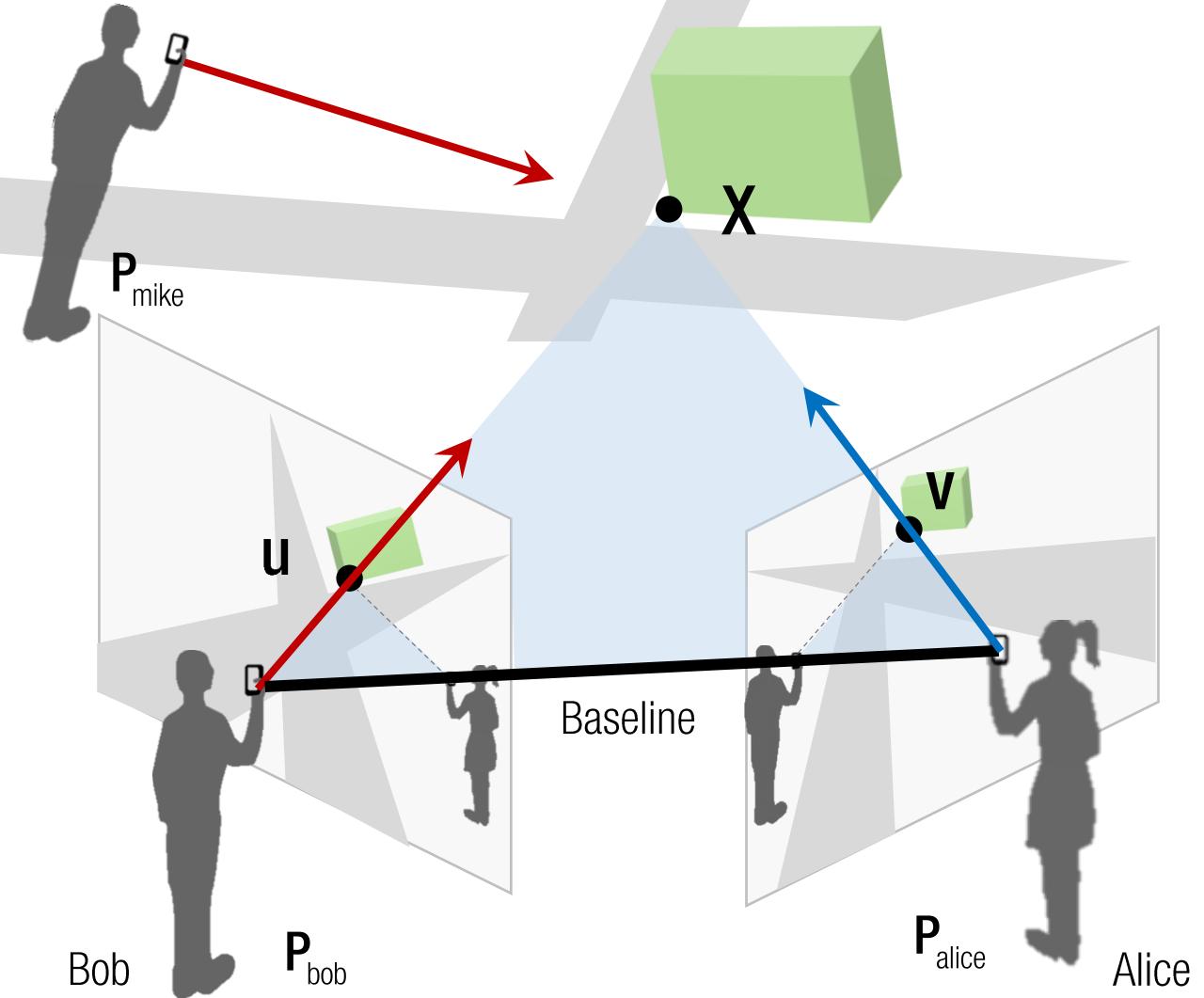
$$\begin{bmatrix} v \\ 1 \end{bmatrix} \times P_{alice} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$

$$\begin{bmatrix} w \\ 1 \end{bmatrix} \times P_{mike} \begin{bmatrix} X \\ 1 \end{bmatrix} = 0$$



- : Knowns
- : Unknowns

Algebraic Error



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

Algebraic error does not have geometric meaning.

Recall: Geometric Verification via Reprojection Error

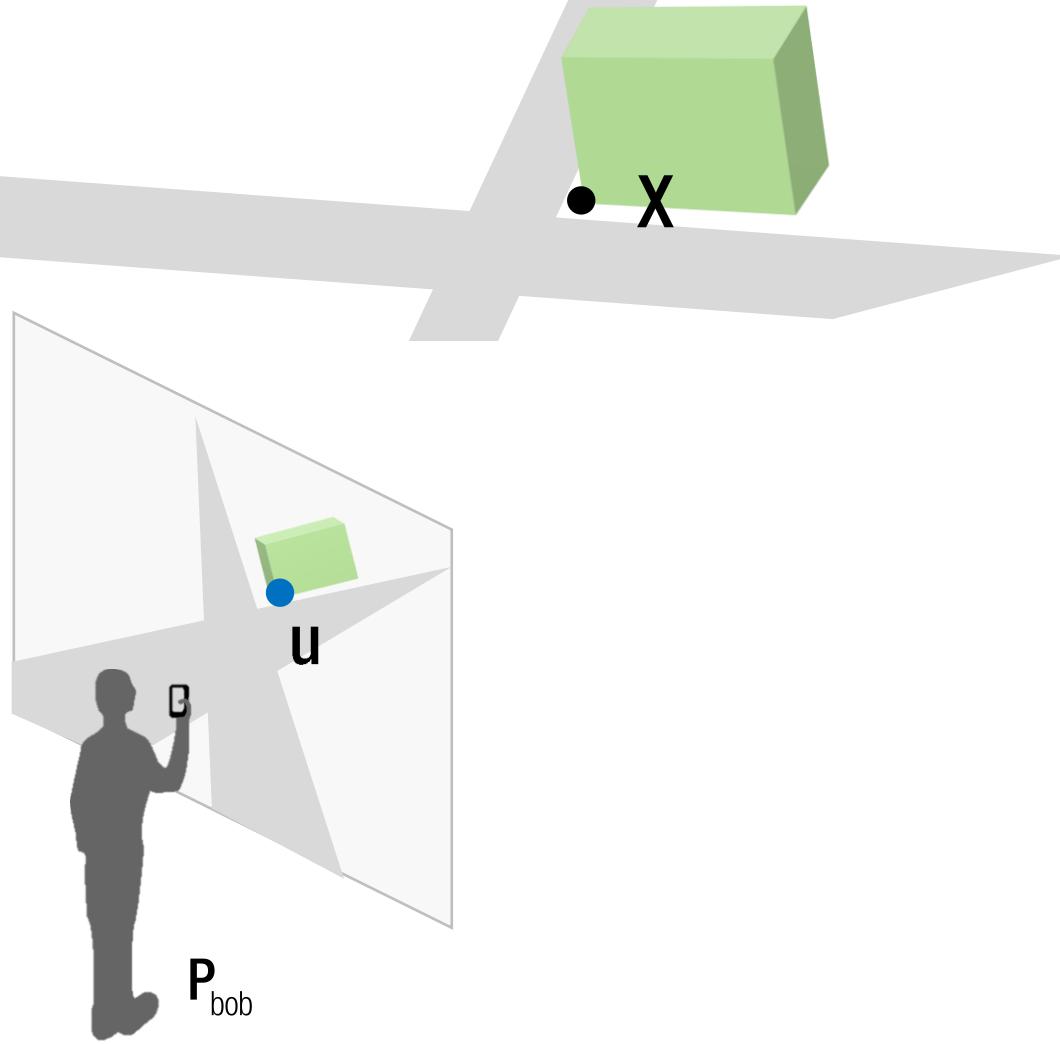


Image feature measurement, e.g. SIFT detection:

u

Recall: Geometric Verification via Reprojection Error

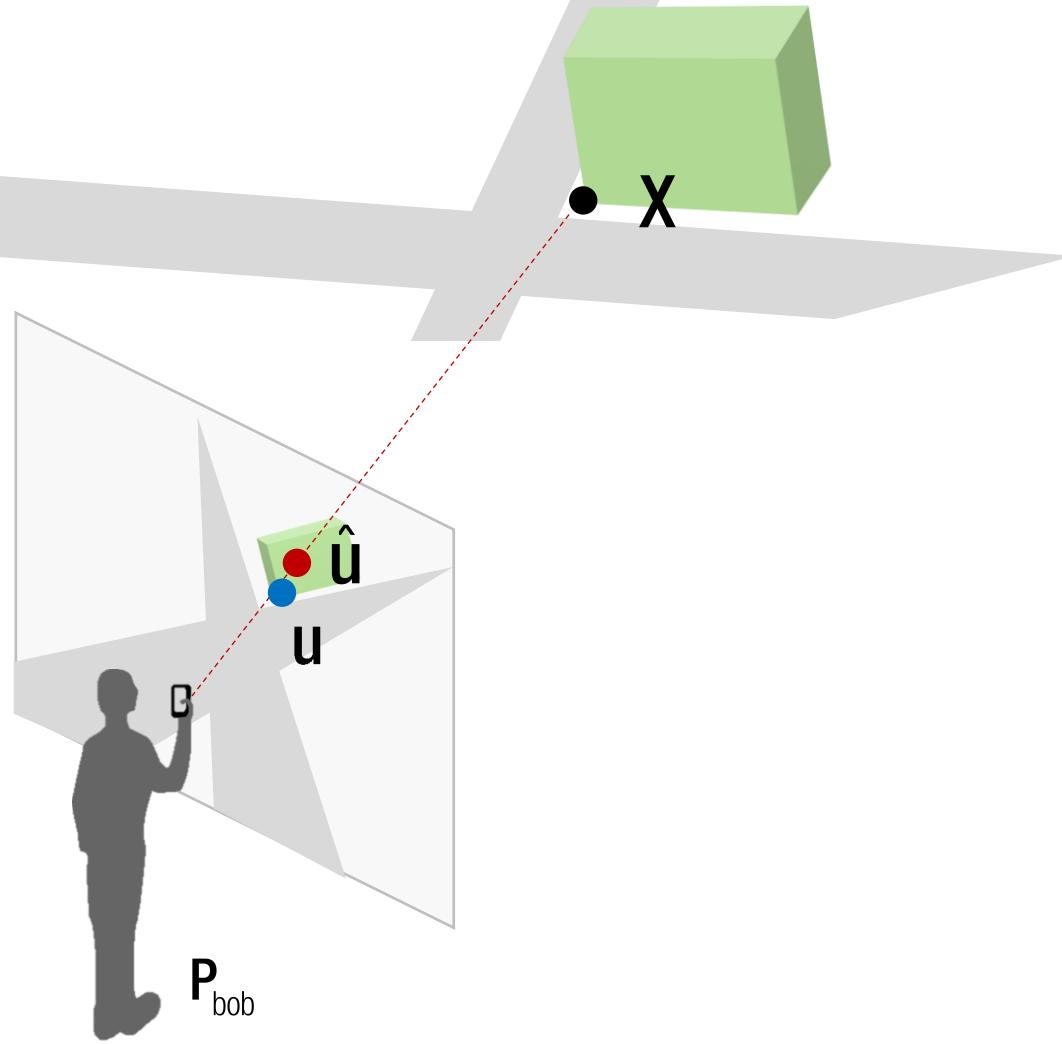


Image feature measurement, e.g. SIFT detection:

$$\mathbf{u}$$

3D point projection, or reprojection:

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

Recall: Geometric Verification via Reprojection Error

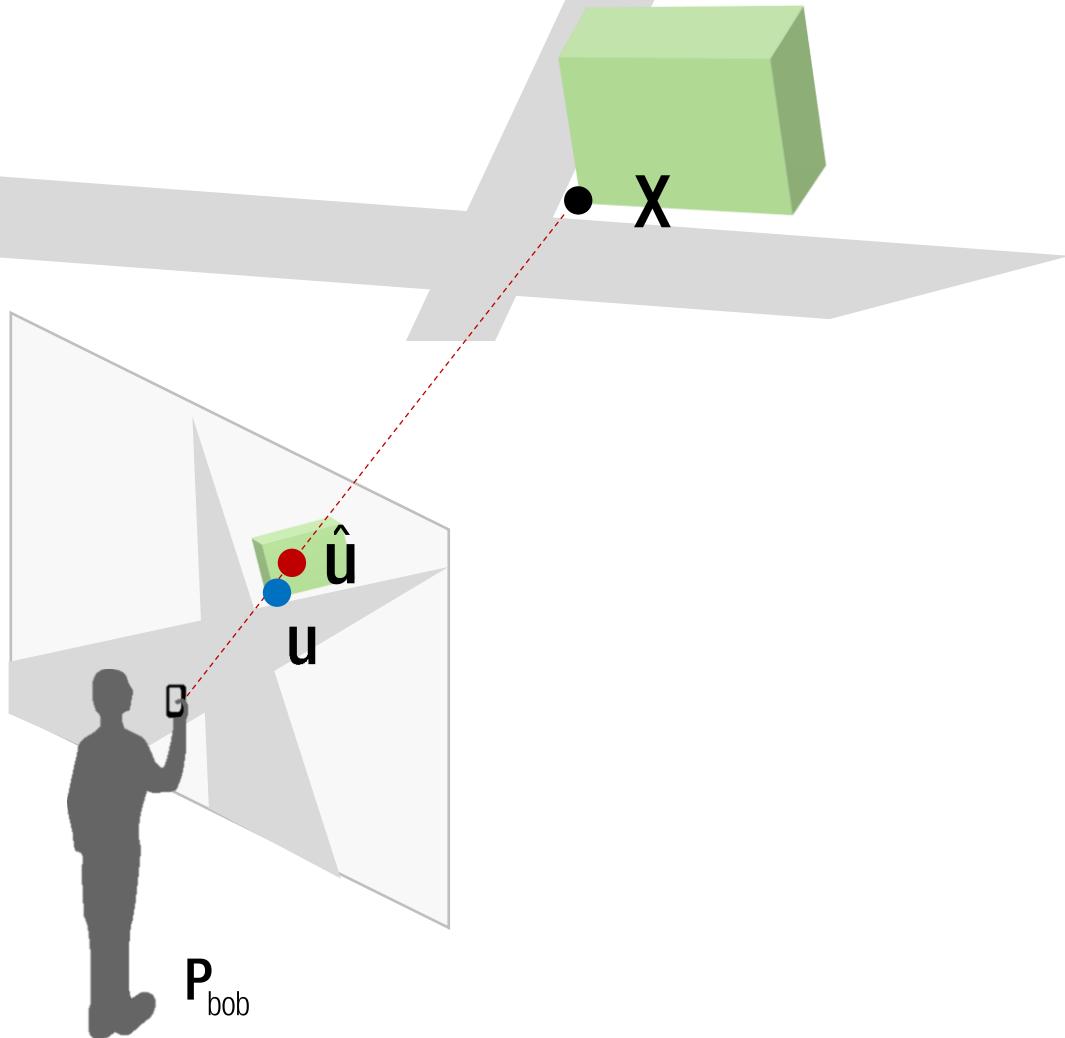


Image feature measurement, e.g. SIFT detection:

$$\mathbf{u}$$

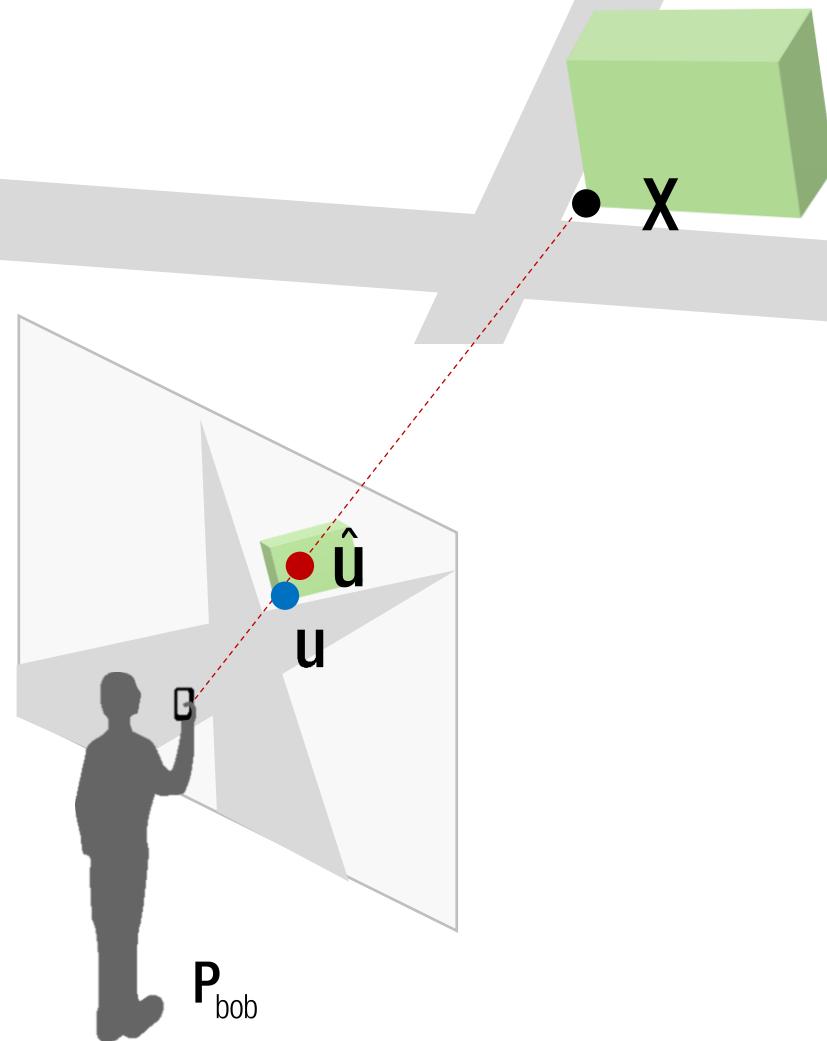
3D point projection, or reprojection:

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

Reprojection error (geometric error):

$$\begin{aligned} E_{\text{geom}} &= \|\hat{\mathbf{u}} - \mathbf{u}\|^2 \\ &= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - \mathbf{u}_1 \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - \mathbf{u}_2 \right)^2 \end{aligned}$$

Algebraic vs. Geometric error

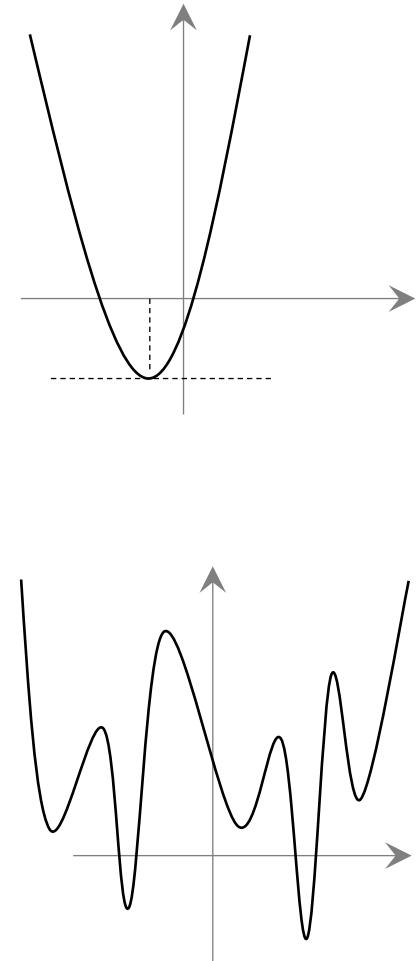


Least squares solution (algebraic error):

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

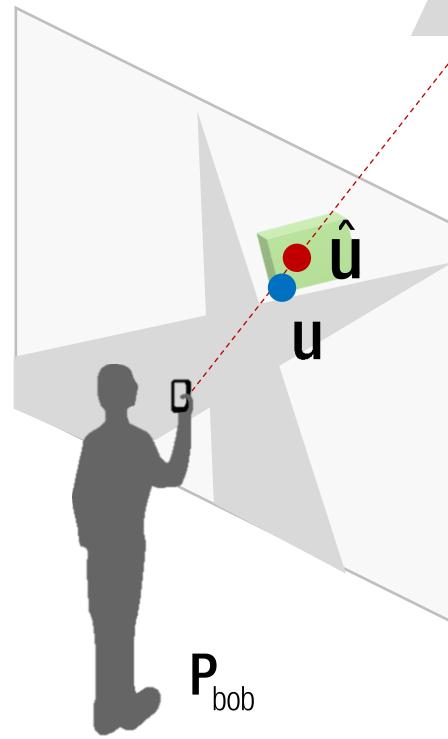
Reprojection error (geometric error):

$$\begin{aligned} E_{\text{geom}} &= \| \hat{\mathbf{u}} - \mathbf{u} \|^2 \\ &= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - \mathbf{u}_1 \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - \mathbf{u}_2 \right)^2 \end{aligned}$$



Black: given variables
Red: unknowns

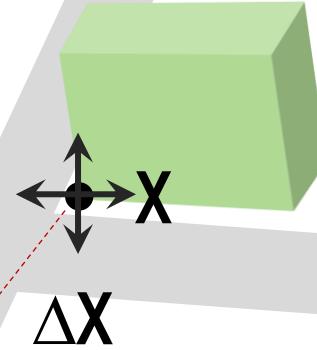
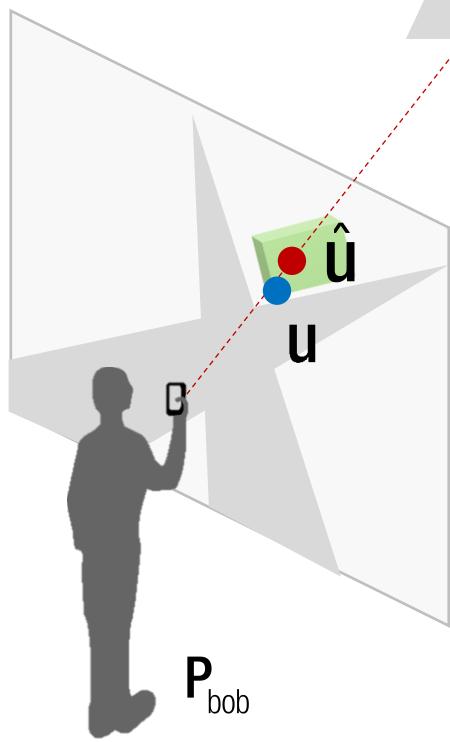
Point Jacobian



$$\begin{aligned} E_{\text{geom}} &= \|\hat{\mathbf{u}} - \mathbf{u}\|^2 \\ &= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - y \right)^2 \end{aligned}$$

Black: given variables
Red: unknowns

Point Jacobian

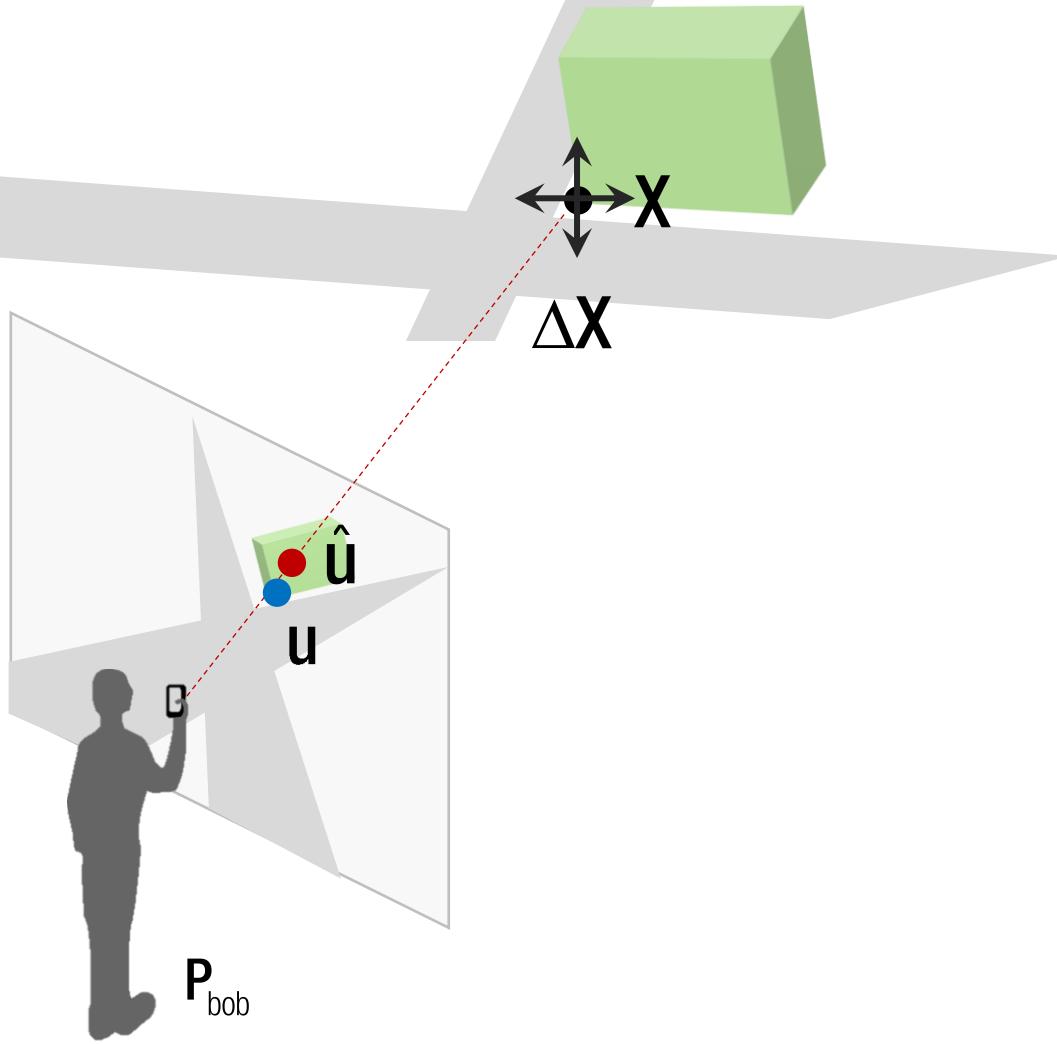


$$\begin{aligned} E_{\text{geom}} &= \|\hat{\mathbf{u}} - \mathbf{u}\|^2 \\ &= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - y \right)^2 \end{aligned}$$

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

Black: given variables
Red: unknowns

Point Jacobian

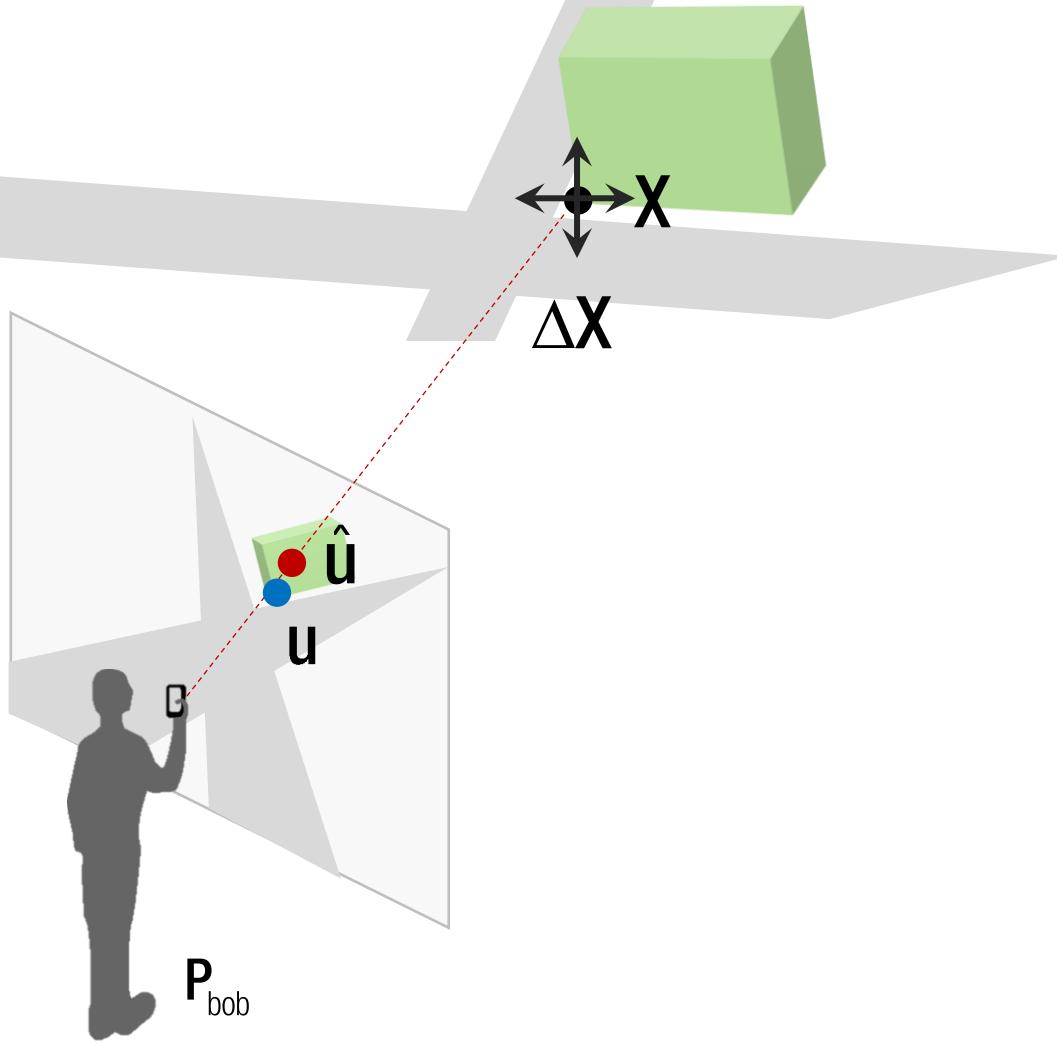


$$E_{\text{geom}} = \left(\frac{u}{w} - x \right)^2 + \left(\frac{v}{w} - y \right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{K}\mathbf{R}(\mathbf{X} - \mathbf{C})$$

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

Black: given variables
Red: unknowns

Point Jacobian



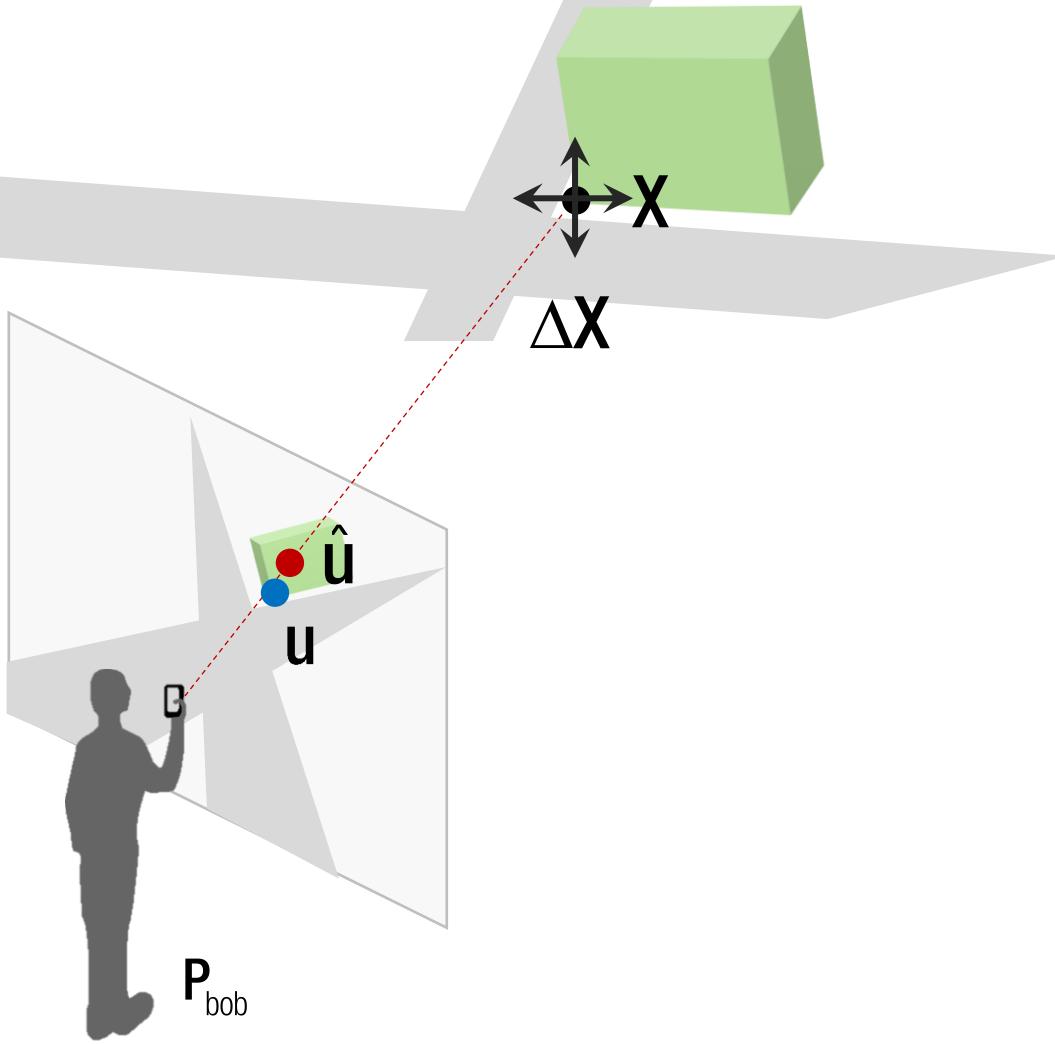
$$E_{\text{geom}} = \left(\frac{u}{w} - x \right)^2 + \left(\frac{v}{w} - y \right)^2 \quad \text{where } \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{K}\mathbf{R}(\mathbf{X} - \mathbf{C})$$

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

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

Black: given variables
Red: unknowns

Point Jacobian



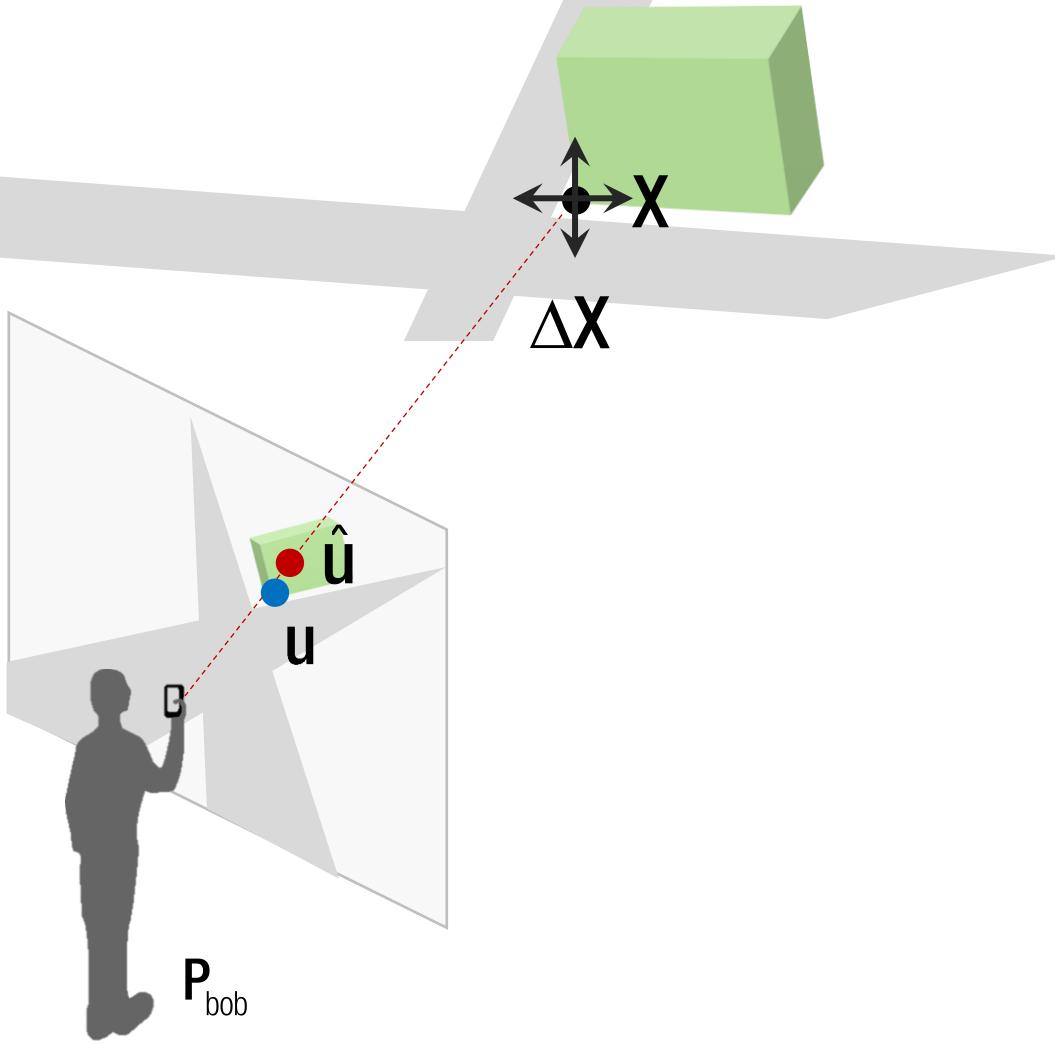
$$E_{\text{geom}} = \left(\frac{u}{w} - x \right)^2 + \left(\frac{v}{w} - y \right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{K}\mathbf{R}(\mathbf{X} - \mathbf{C})$$

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

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

Black: given variables
Red: unknowns

Point Jacobian



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

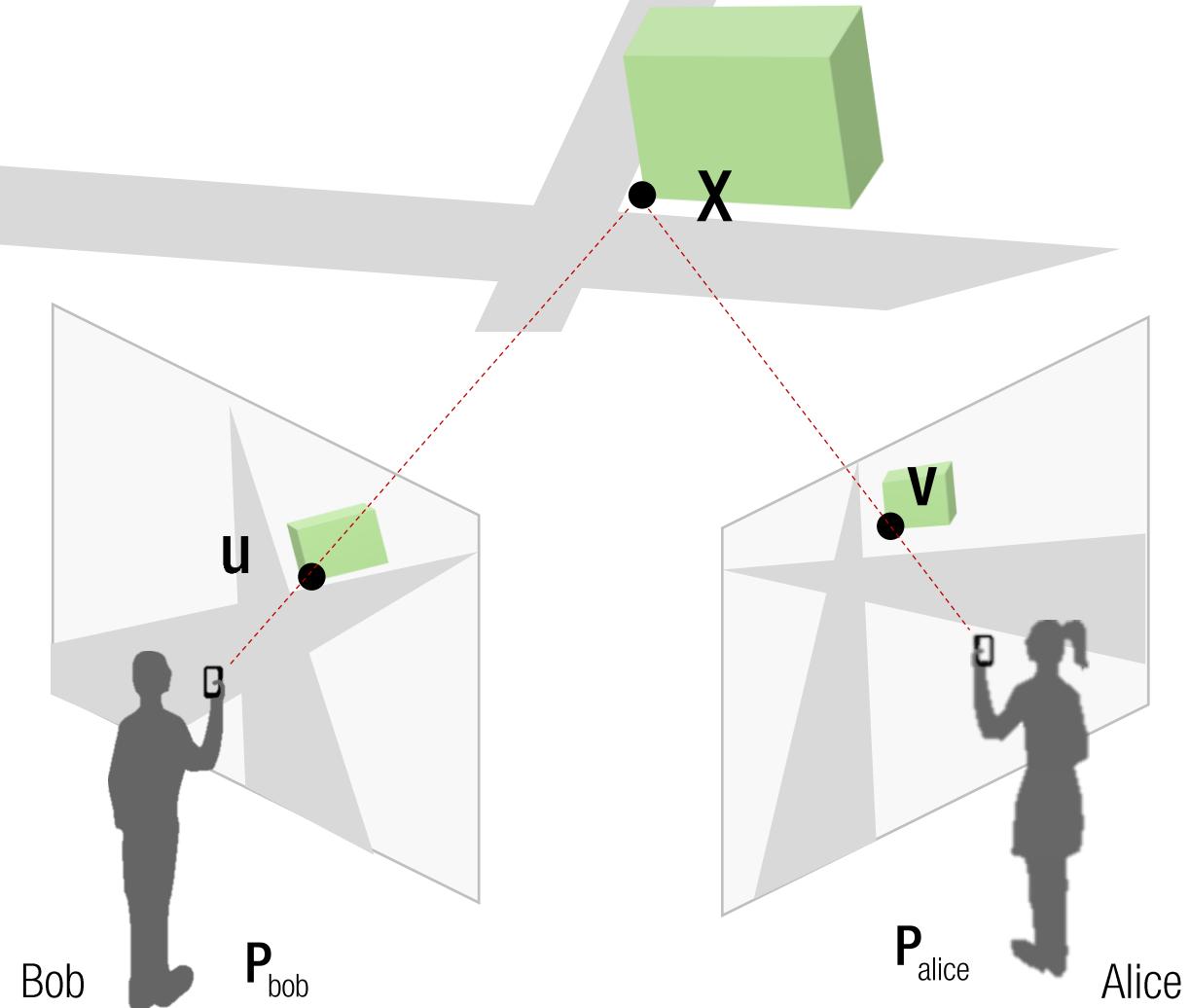
$$\rightarrow \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{K}\mathbf{R}$$

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

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

Black: given variables
Red: unknowns

Point Jacobian



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

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

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

Algorithm 3 Nonlinear Point Refinement

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

$$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial}{\partial \mathbf{X}} \begin{bmatrix} \frac{u}{w} \\ \frac{v}{w} \end{bmatrix} = \begin{bmatrix} w \frac{\partial u}{\partial \mathbf{X}} - u \frac{\partial w}{\partial \mathbf{X}} \\ \frac{w^2}{w^2} \\ v \frac{\partial u}{\partial \mathbf{X}} - v \frac{\partial w}{\partial \mathbf{X}} \\ \frac{w^2}{w^2} \end{bmatrix}$$
$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\top \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\top (\mathbf{b} - f(\mathbf{x}))$$

Algorithm 3 Nonlinear Point Refinement

```

1:  $\mathbf{b} = [\mathbf{u}_1^\top \mathbf{u}_2^\top]^\top$ 
2: for  $j = 1 : n\text{Iters}$  do
3:   Build point Jacobian,  $\frac{\partial f(\mathbf{X})_j}{\partial \mathbf{X}}$ .
4:   Compute  $f(\mathbf{X})$ .
5:    $\Delta \mathbf{X} = \left( \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}^\top \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} + \boxed{\lambda \mathbf{I}} \right)^{-1} \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}^\top (\mathbf{b} - f(\mathbf{X}))$ 
6:    $\mathbf{X} = \mathbf{X} + \Delta \mathbf{X}$ 
7: end for

```

Damping factor (Levenberg-Marquardt algorithm)

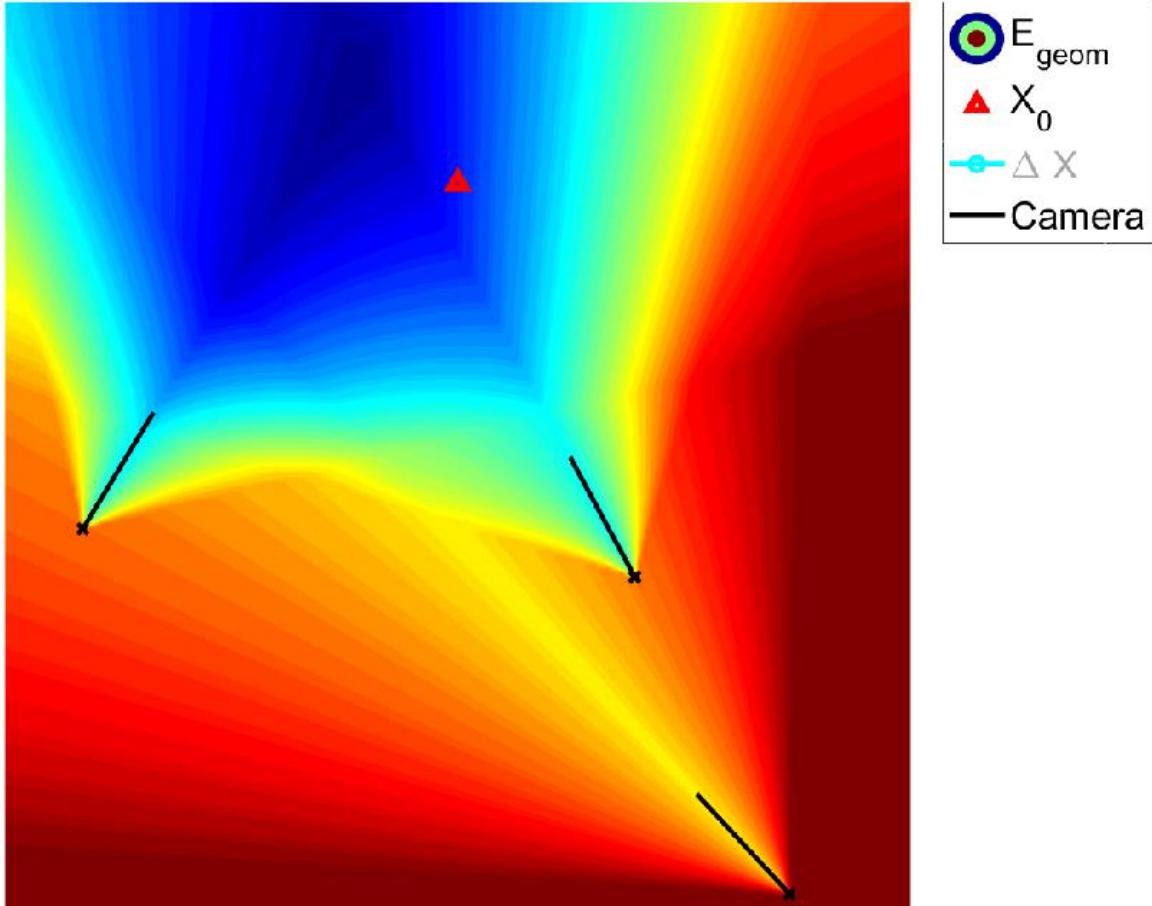
$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\top \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} + \lambda \mathbf{I} \right)^{-1} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}^\top (\mathbf{b} - f(\mathbf{x}))$$

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

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

Black: given variables
Red: unknowns

Example: 1D Camera Triangulation

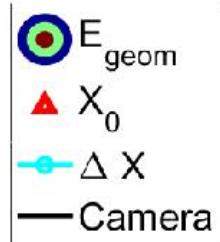
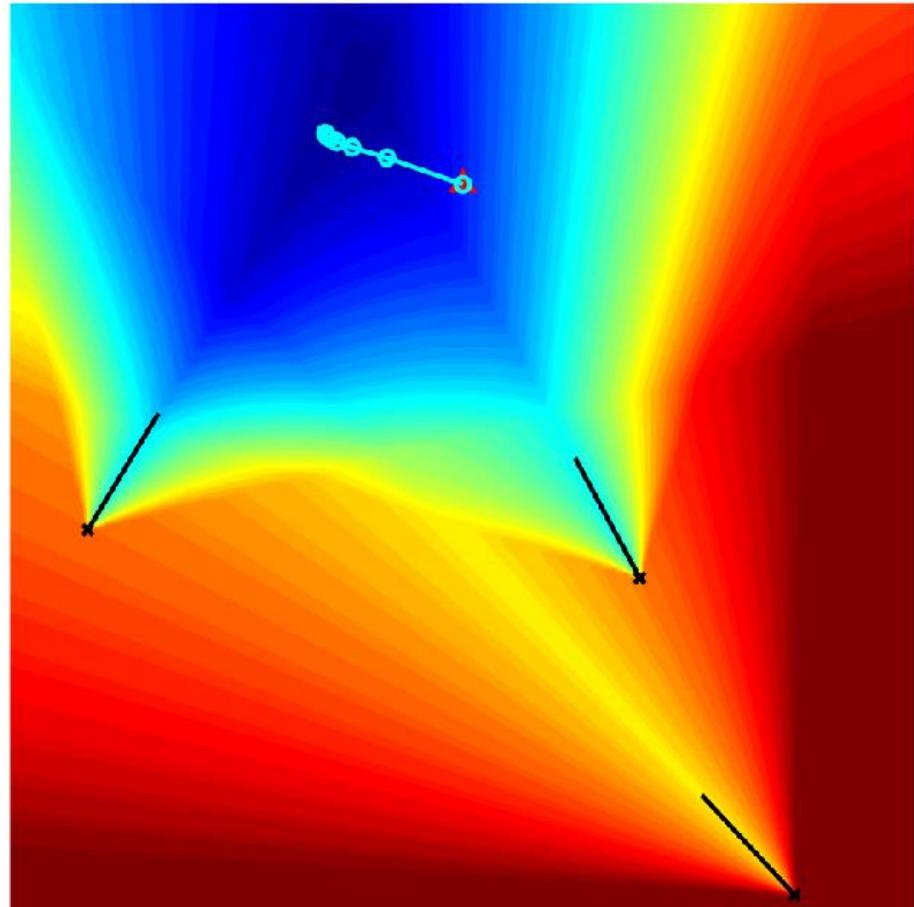


```
function df_dx = JacobianX_1D(K, R, C, X)  
  
x = K * R * (X-C);  
  
u = x(1);  
w = x(2);  
  
del = K * R;  
du_dc = del(1,:);  
dw_dc = del(2,:);  
  
df_dx = [(w*du_dc-u*dw_dc)/w^2];
```

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

Black: given variables
Red: unknowns

Example: 1D Camera Triangulation



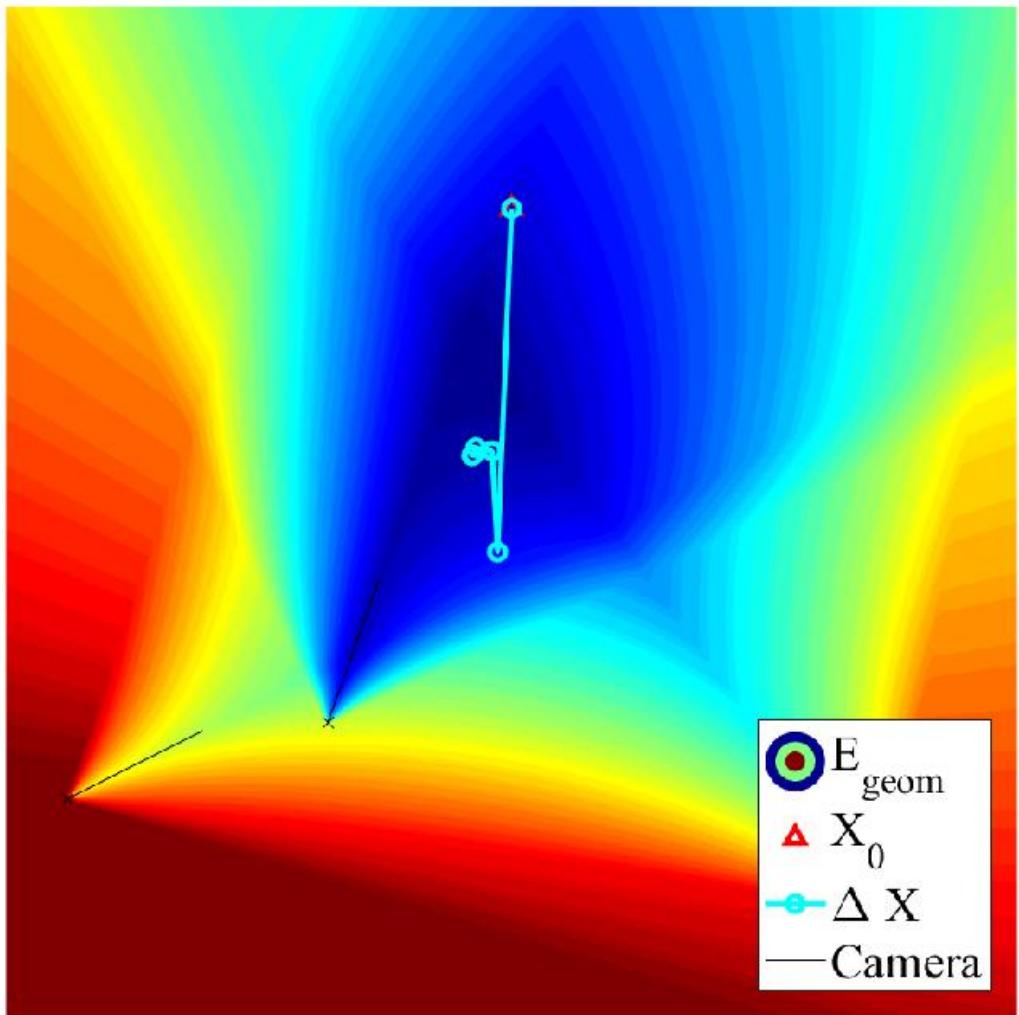
```
for j = 1 : 10
    df_dX = [];
    delta_b = [];
    for i = 1 : size(c,2)
        df_dX = [df_dX; JacobianX(eye(2), R{i}, c(:,i), x)];
        u = R{i} * (x-c(:,i));
        delta_b = [delta_b; -u(1)/u(2)];
    end

    jacobian = df_dX;
    norm(delta_b)

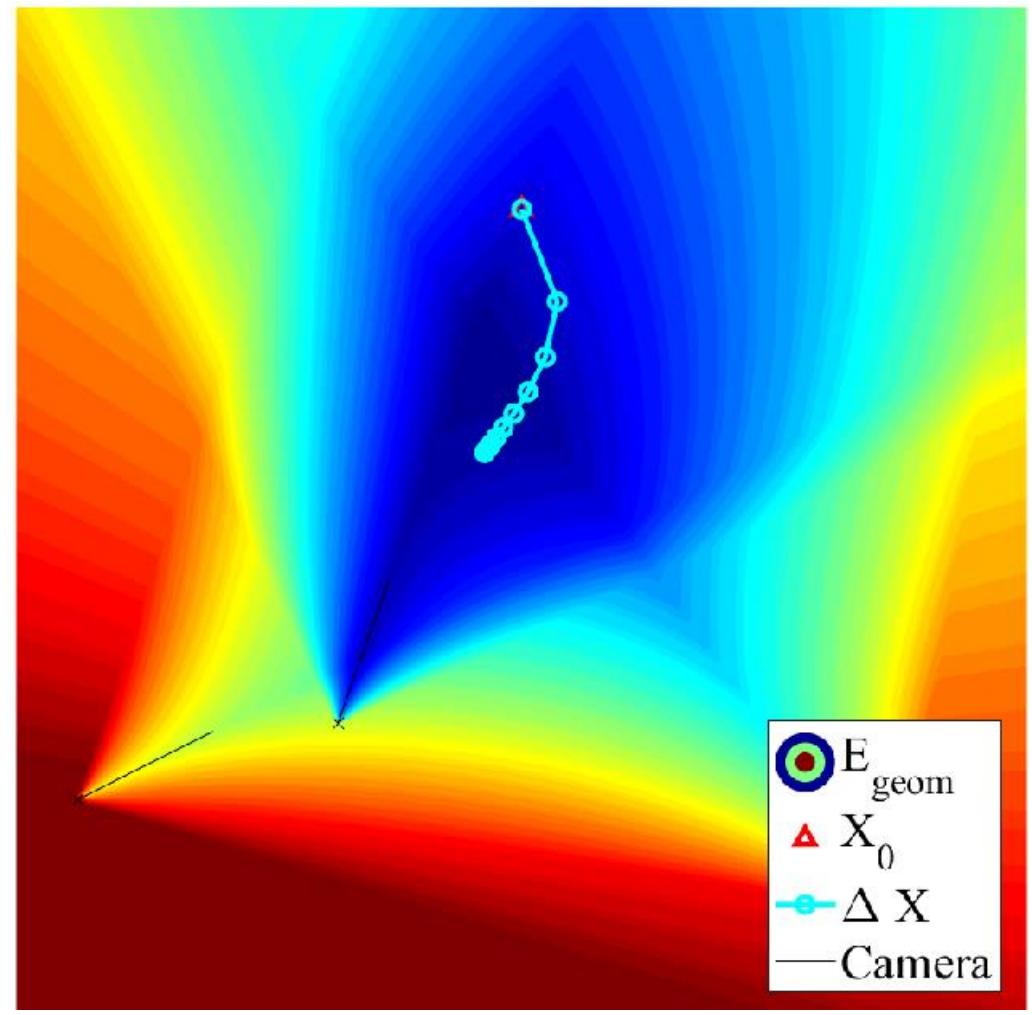
    delta_x =
    inv(jacobian'*jacobian+lambda*eye(size(jacobian'*jacobian,1)))*jacobian'*delta_b;
    x = x + delta_x;
    X(:,j+1) = x;
end
```

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

Damping Factor



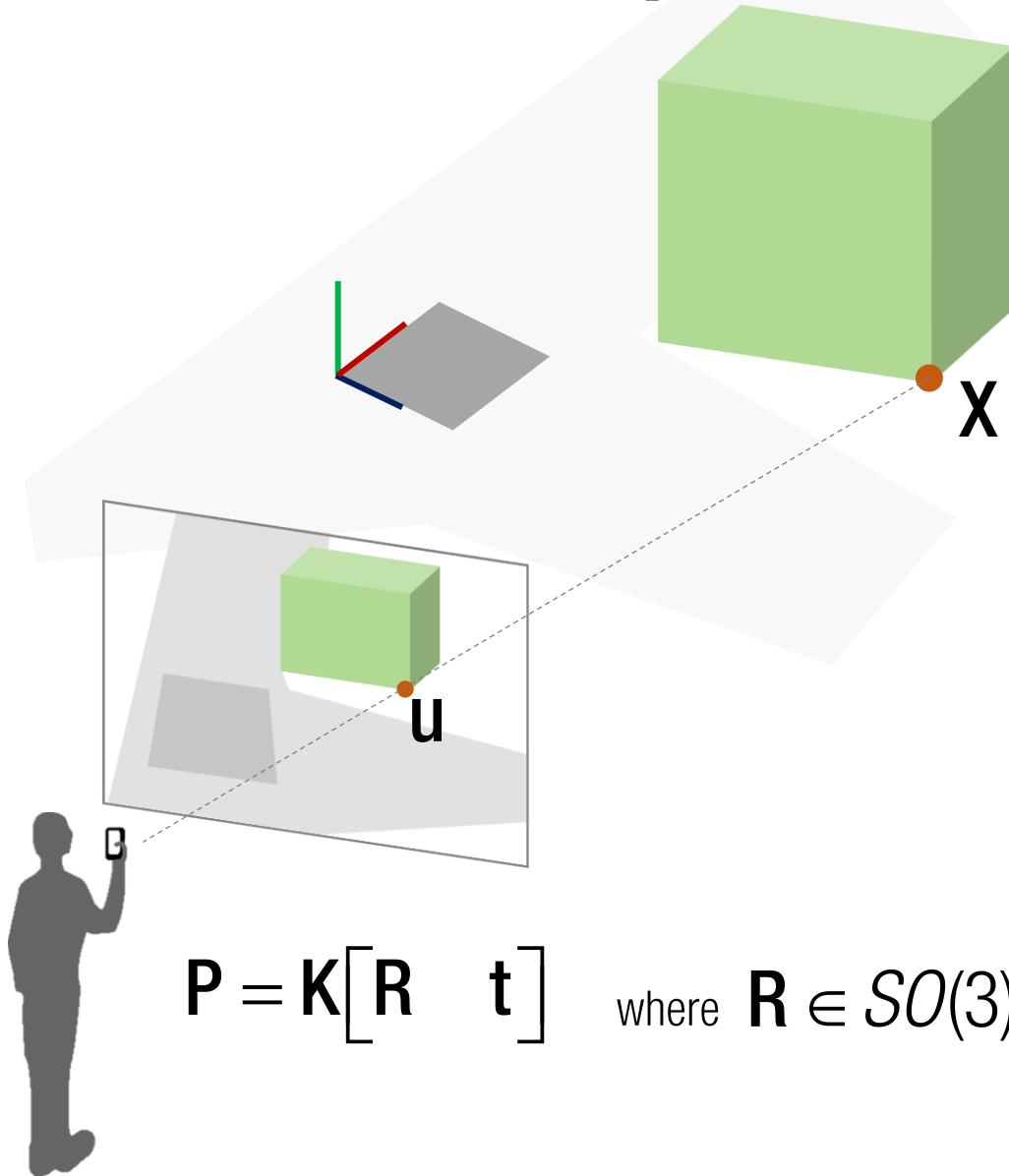
$$\Delta \mathbf{x} = \left(\frac{\partial f(\mathbf{x})^\top}{\partial \mathbf{x}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right)^{-1} \frac{\partial f(\mathbf{x})^\top}{\partial \mathbf{x}} (\mathbf{b} - f(\mathbf{x}))$$



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

Pose Refinement

3D-2D Correspondence

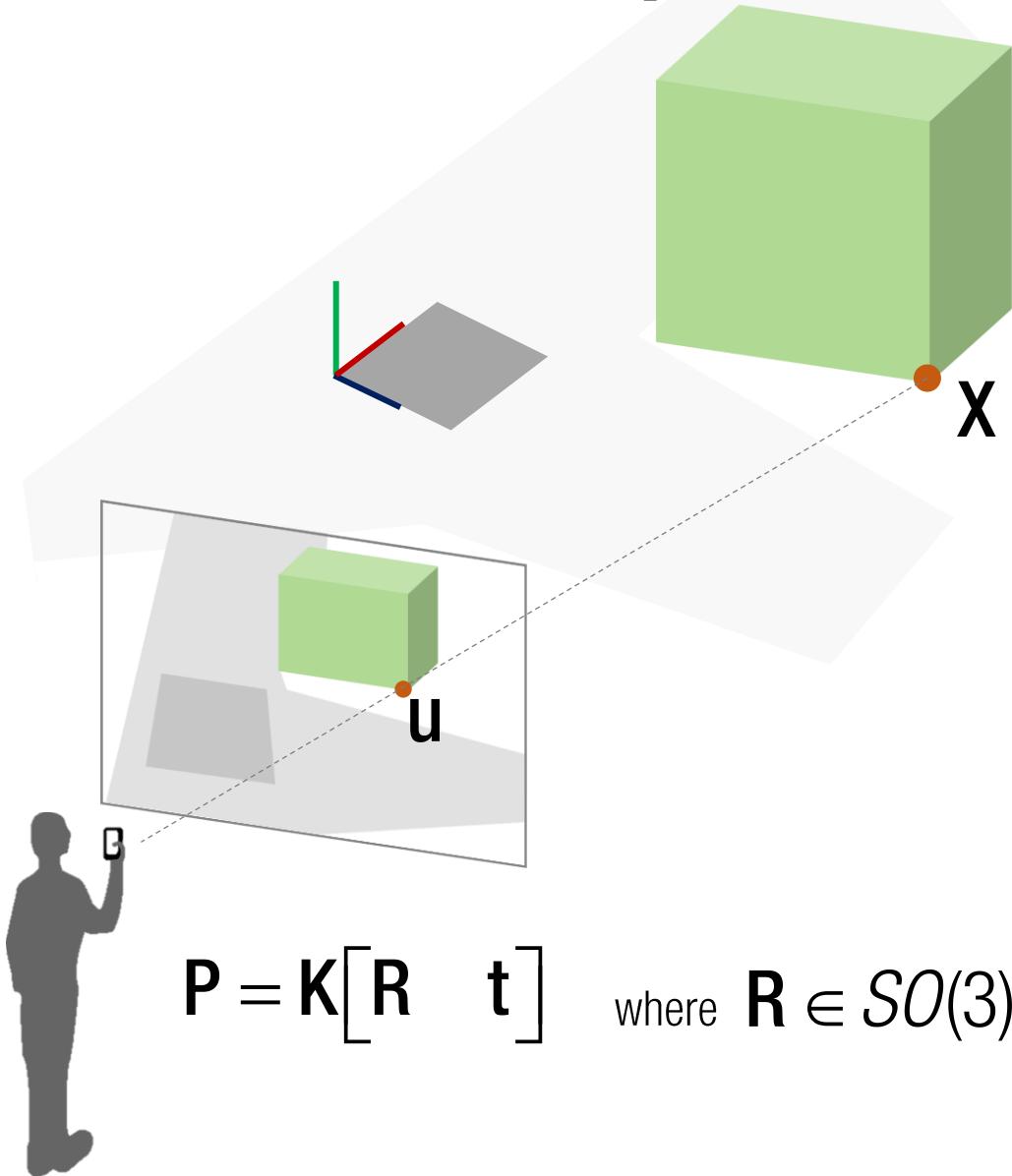


3D-2D correspondence: $u \leftrightarrow X$

$$\lambda u = K[R \ t]X$$

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

3D-2D Correspondence



$$P = K[R \ t] \quad \text{where } R \in SO(3)$$

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

$$\lambda u = K[R \ t]X$$

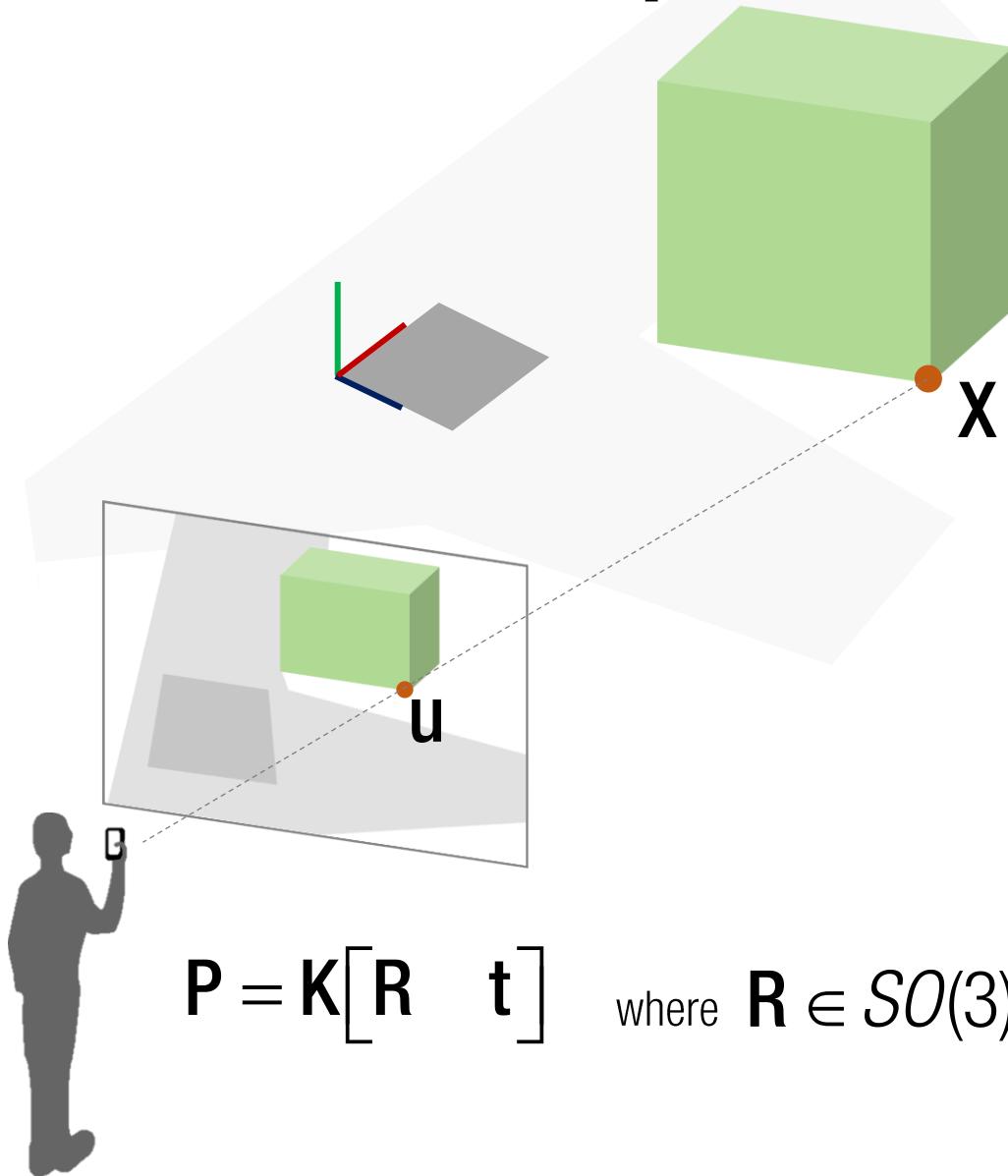
$$\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Linear in camera matrix

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

3D-2D Correspondence

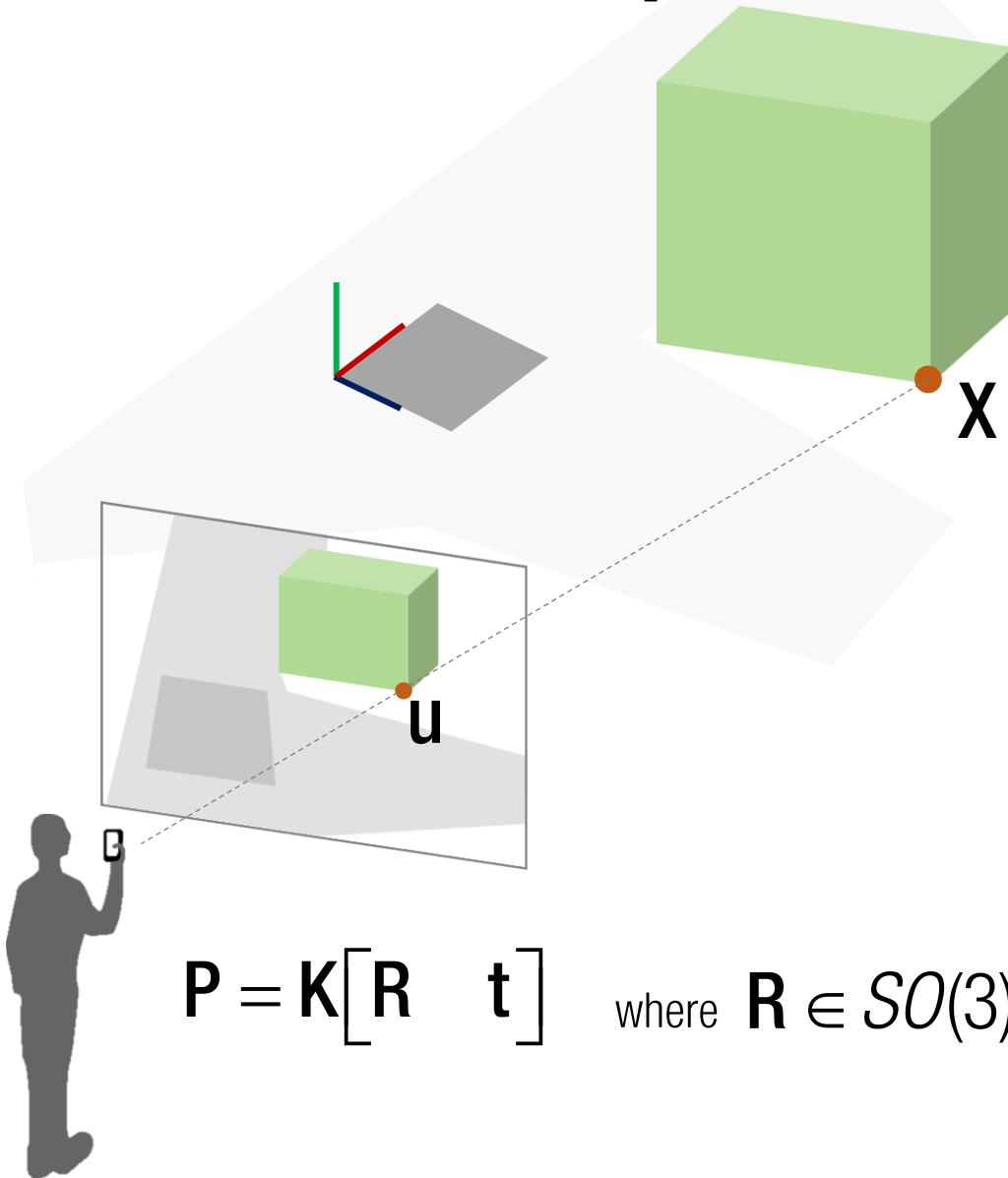


3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

3D-2D Correspondence



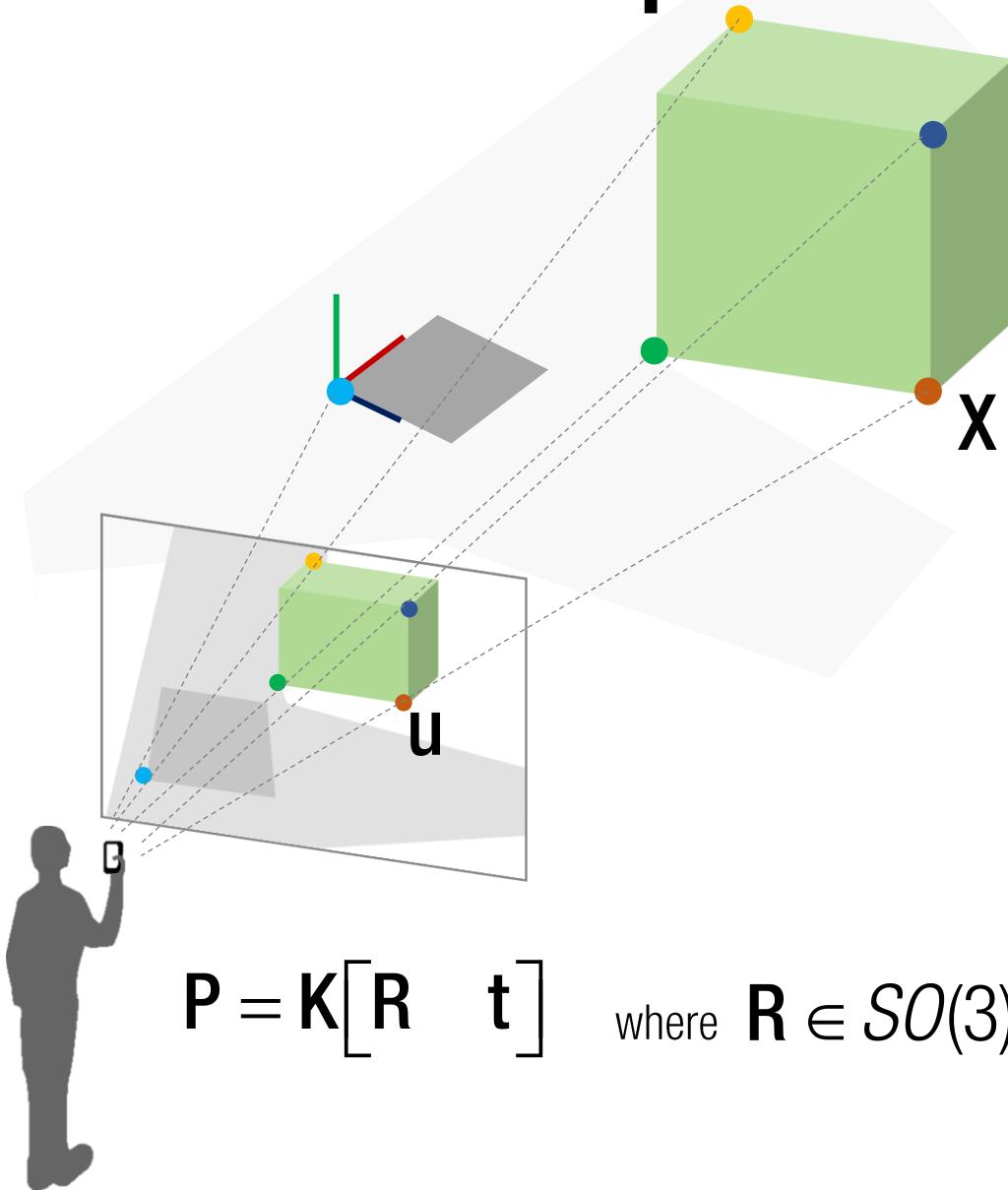
3D-2D correspondence: $\mathbf{u} \leftrightarrow \mathbf{X}$

$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$\begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{bmatrix} = \begin{bmatrix} X & Y & Z & 1 & -u^x X & -u^x Y & -u^x Z & -u^x \\ & X & Y & Z & 1 & -u^y X & -u^y Y & -u^y Z & -u^y \end{bmatrix}_{2 \times 12} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

3D-2D Correspondence



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

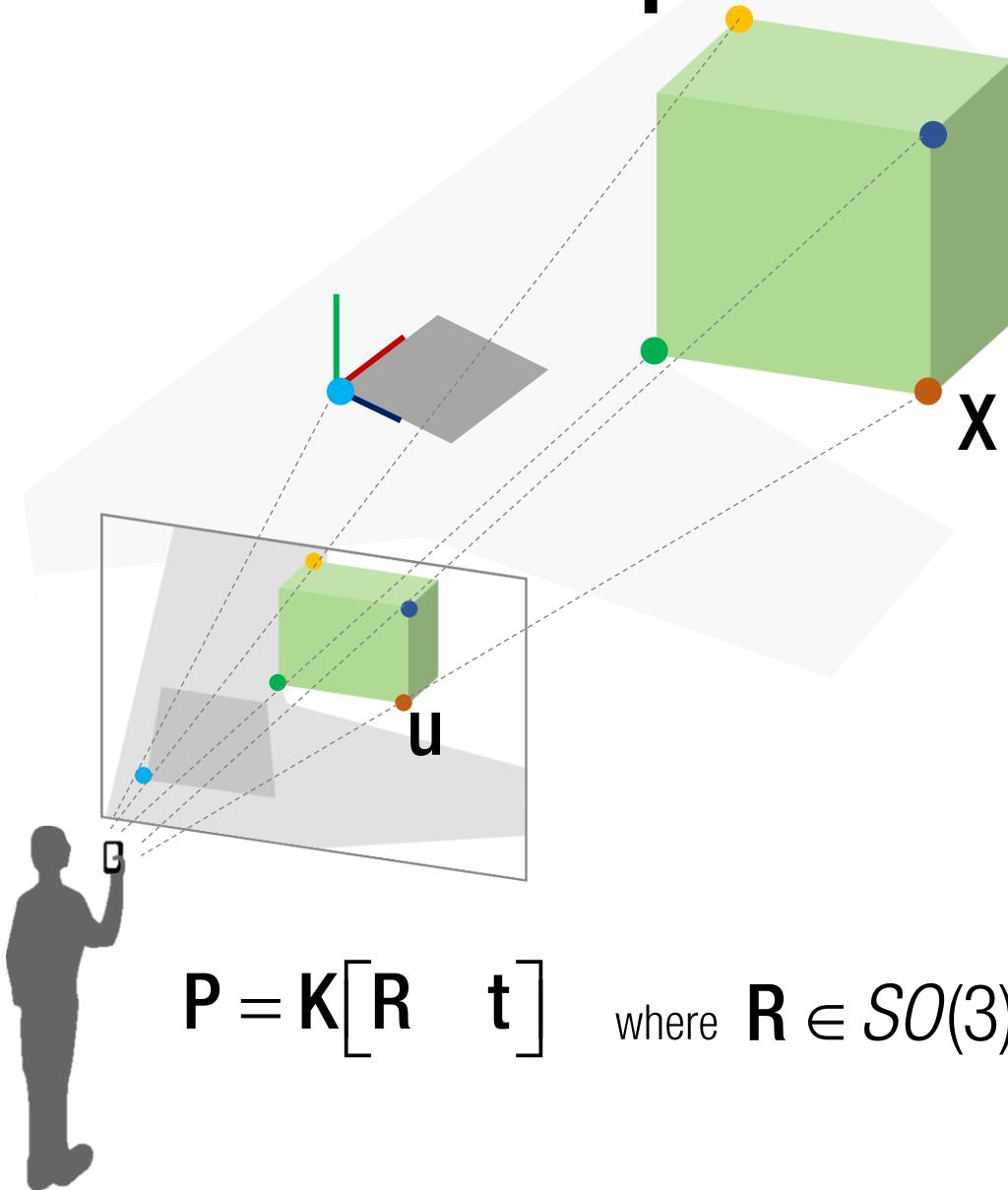
$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

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

$2m \times 12$

3D-2D Correspondence



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

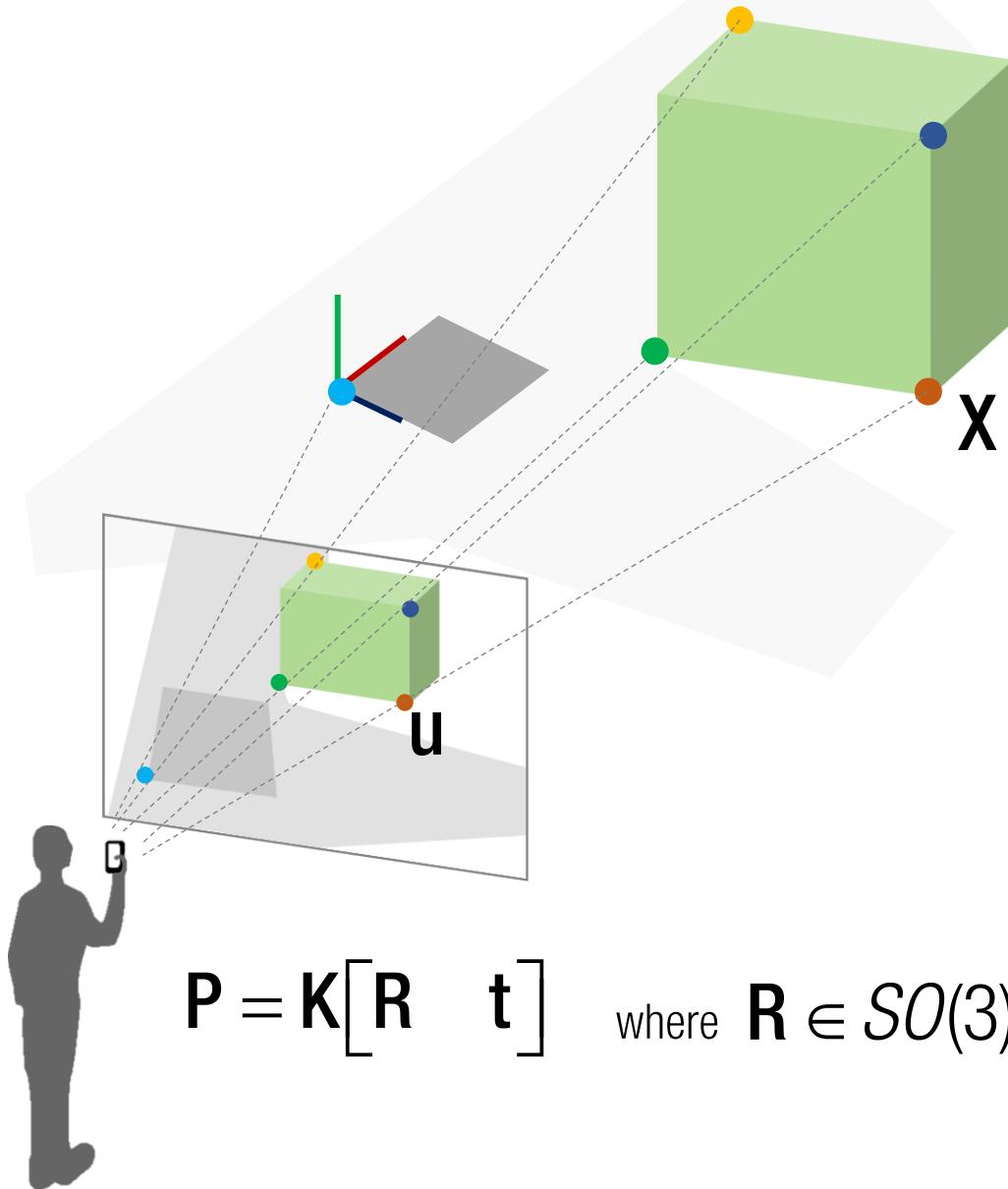
$$u^x = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$u^y = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

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

$2m \times 12$

Camera Pose Estimation



If K is given,

$$K[R \ t] = \gamma [p_1 \ p_2 \ p_3 \ p_4]$$

$$\rightarrow \gamma R = K^{-1} [p_1 \ p_2 \ p_3]$$

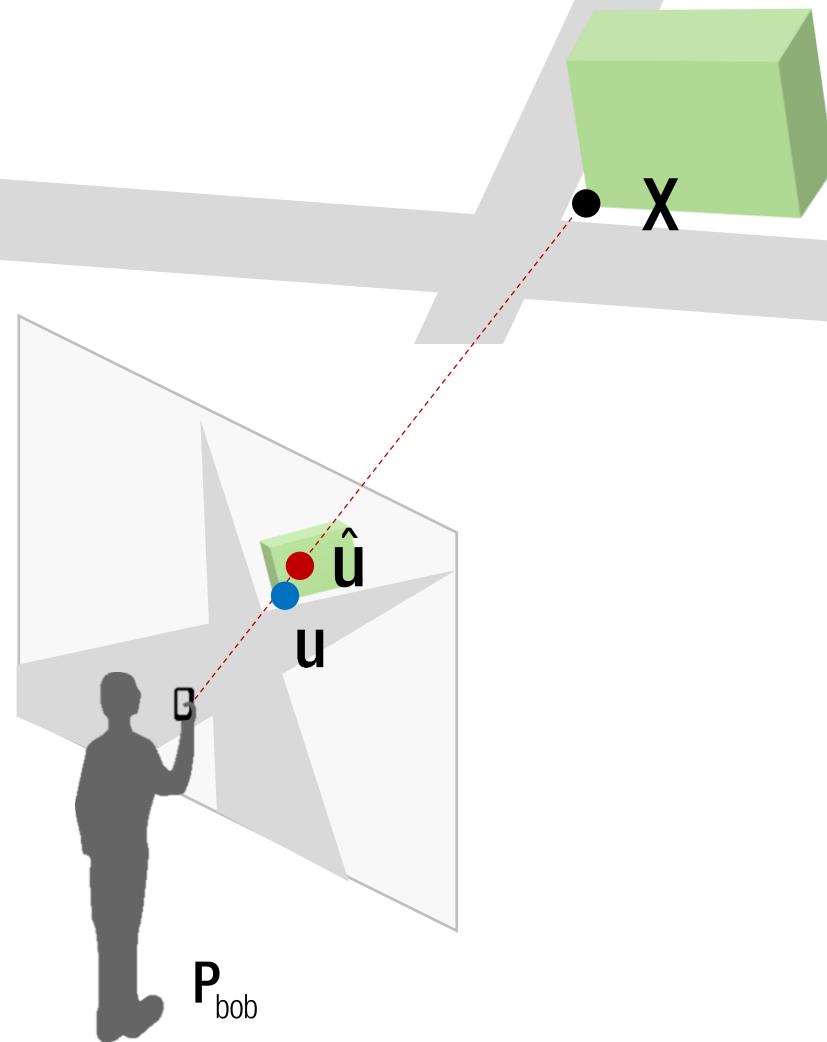
$$K^{-1} [p_1 \ p_2 \ p_3] = U \begin{bmatrix} d_{11} & & \\ & d_{22} & \\ & & d_{33} \end{bmatrix} V^T$$

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

$$R = UV^T \quad : \text{SVD cleanup}$$

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

Algebraic vs. Geometric error

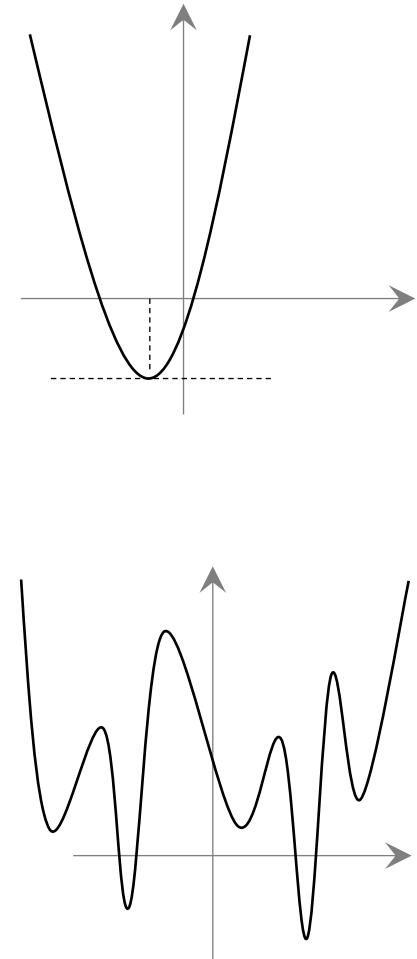


Least squares solution (algebraic error):

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

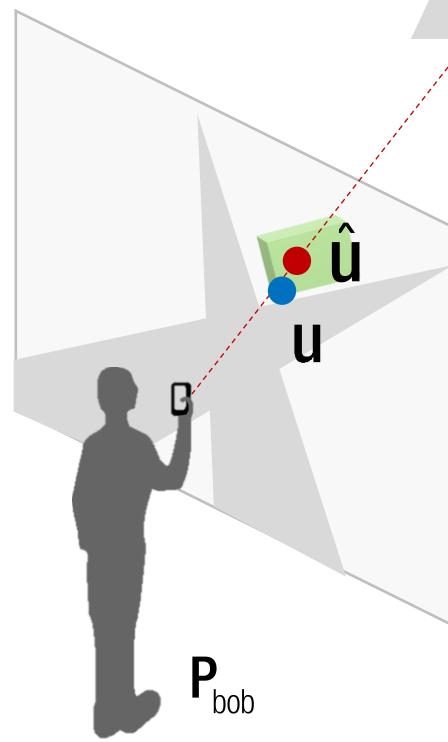
Reprojection error (geometric error):

$$\begin{aligned} E_{\text{geom}} &= \| \hat{\mathbf{u}} - \mathbf{u} \|^2 \\ &= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - \mathbf{u}_1 \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - \mathbf{u}_2 \right)^2 \end{aligned}$$



Black: given variables
Red: unknowns

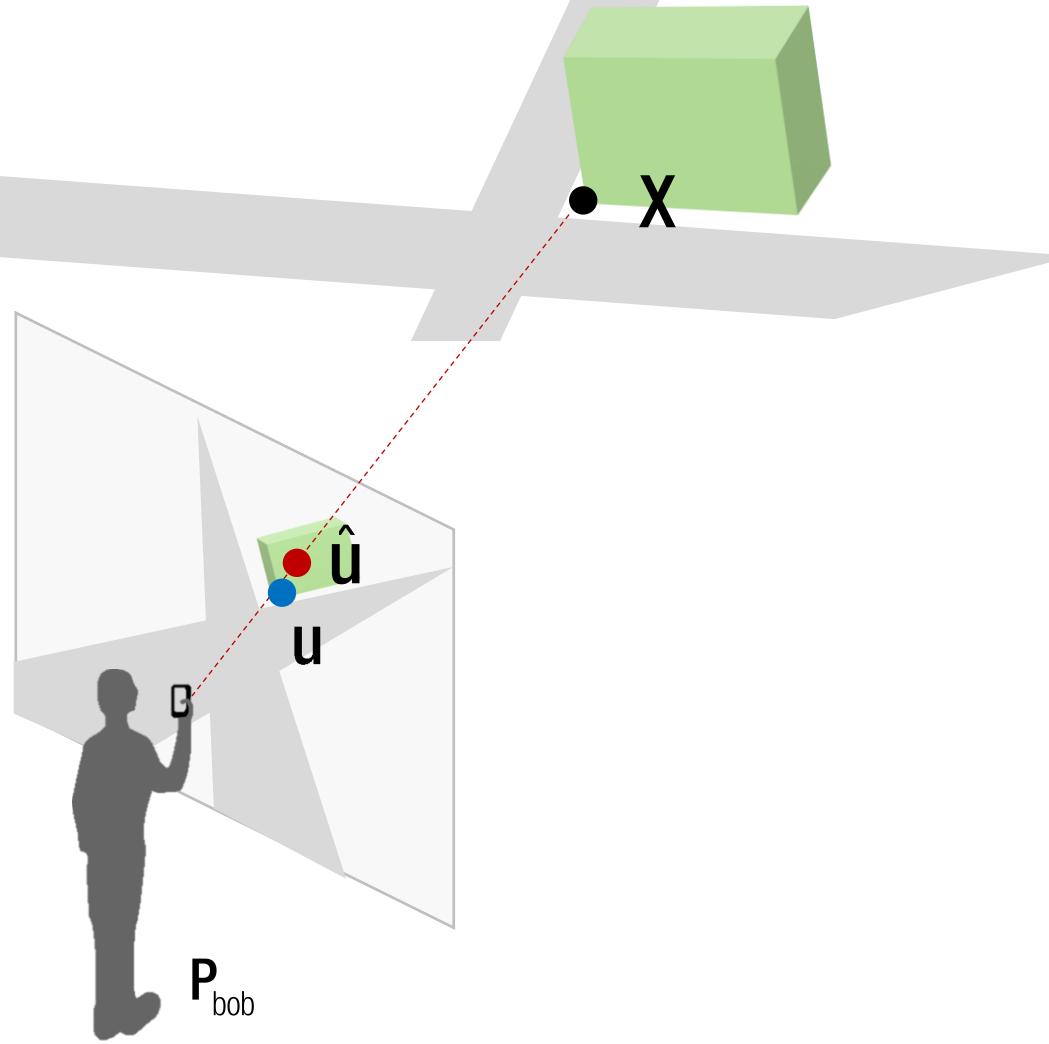
Point Jacobian



$$\begin{aligned} E_{\text{geom}} &= \|\hat{\mathbf{u}} - \mathbf{u}\|^2 \\ &= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - y \right)^2 \end{aligned}$$

Black: given variables
Red: unknowns

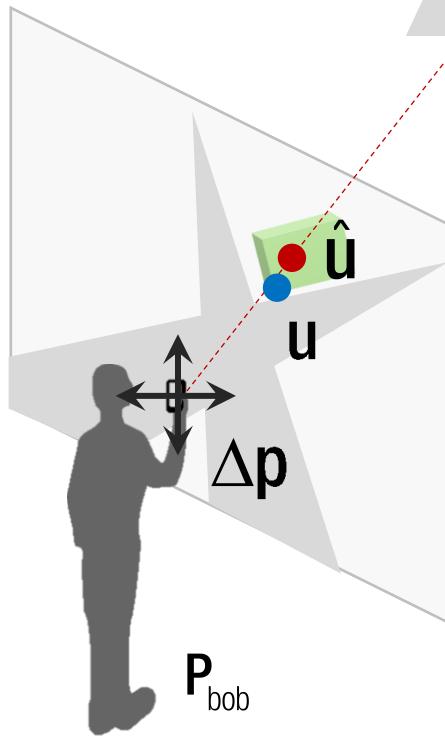
Camera Jacobian



$$\begin{aligned} E_{\text{geom}} &= \|\hat{\mathbf{u}} - \mathbf{u}\|^2 \\ &= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - y \right)^2 \end{aligned}$$

Black: given variables
Red: unknowns

Camera Jacobian

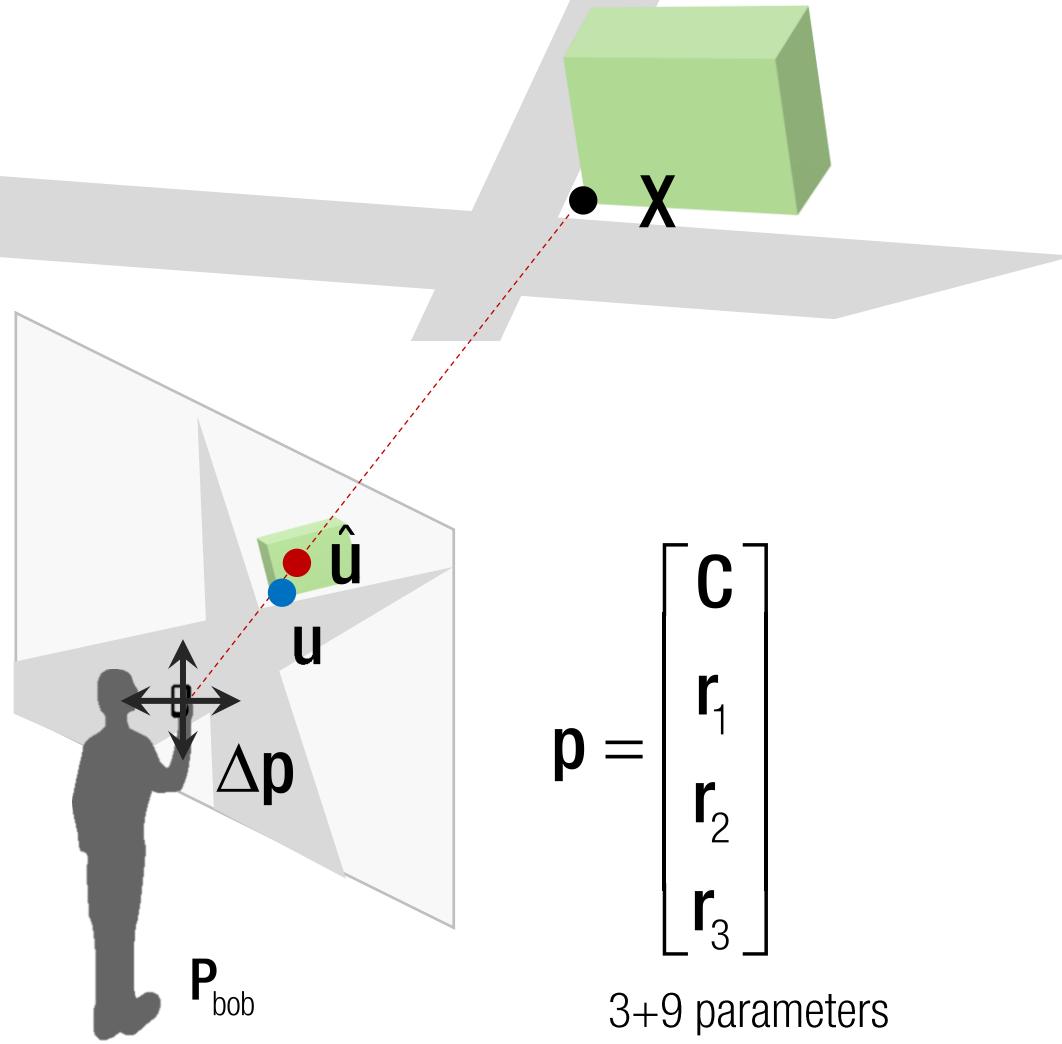


$$\begin{aligned}
 E_{\text{geom}} &= \|\hat{\mathbf{u}} - \mathbf{u}\|^2 \\
 &= \left(\frac{\mathbf{P}_1 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - x \right)^2 + \left(\frac{\mathbf{P}_2 \mathbf{X}}{\mathbf{P}_3 \mathbf{X}} - y \right)^2
 \end{aligned}$$

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

Black: given variables
Red: unknowns

Camera Jacobian



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

3+9 parameters

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

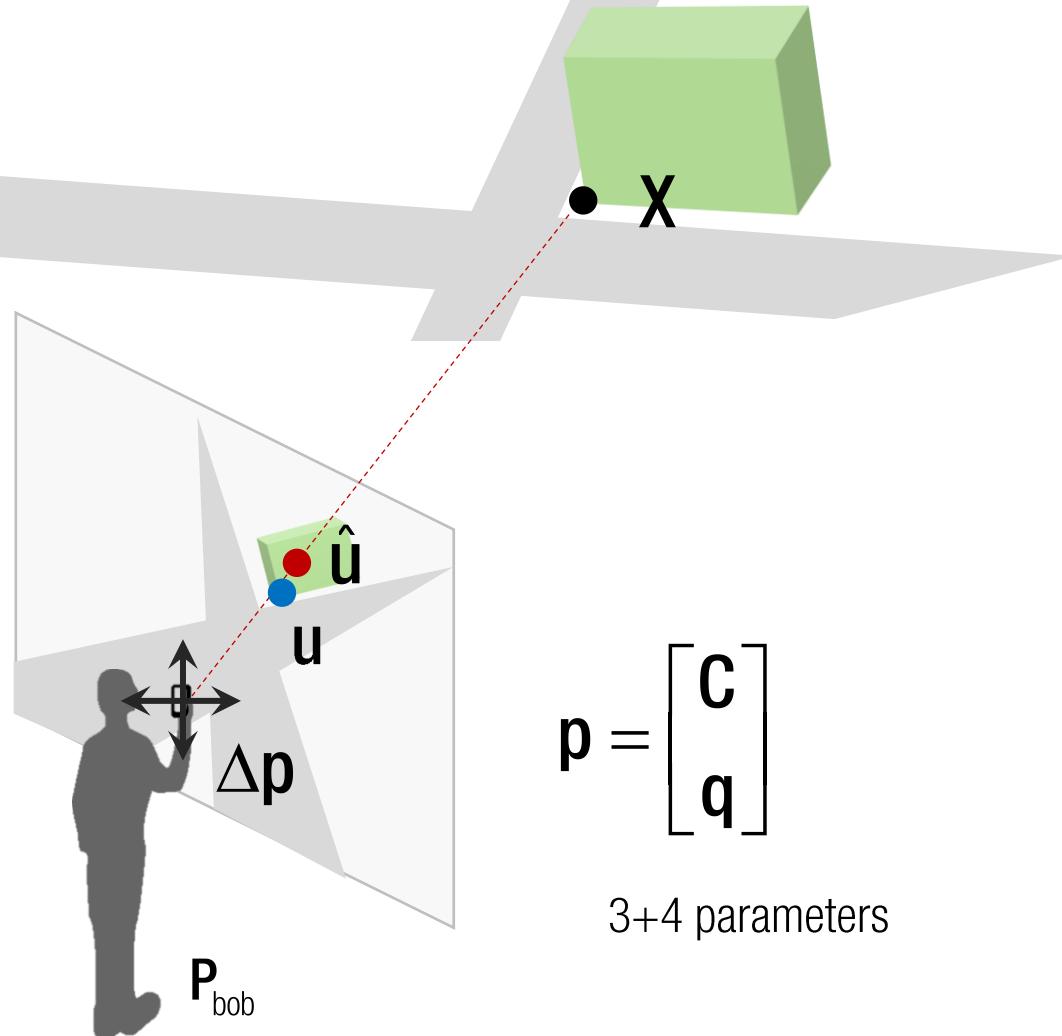
$$\rightarrow \frac{\partial}{\partial \mathbf{C}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = -\mathbf{KR}$$

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

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

Black: given variables
Red: unknowns

Camera Jacobian



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

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

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

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

: Chain rule

Quaternion jacobian

Quaternion Jacobian

$$\mathbf{R} = \begin{bmatrix} 1 - 2q_y^2 - 2q_z^2 & 2q_x q_y - 2q_z q_w & 2q_x q_z + 2q_y q_w \\ 2q_x q_y + 2q_z q_w & 1 - 2q_x^2 - 2q_z^2 & 2q_y q_z - 2q_x q_w \\ 2q_x q_z - 2q_y q_w & 2q_y q_z + 2q_x q_w & 1 - 2q_x^2 - 2q_y^2 \end{bmatrix}$$

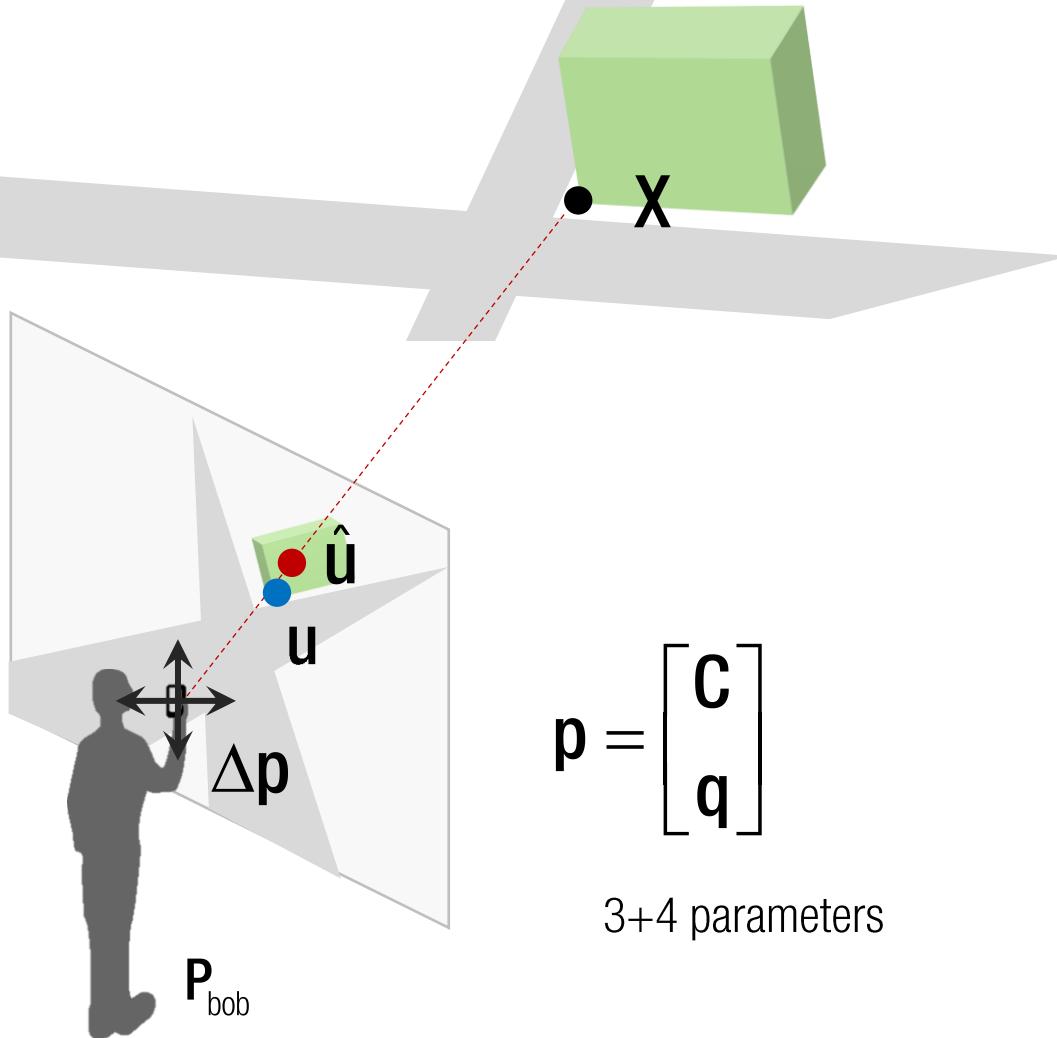
where

$$\mathbf{q} = [q_w \quad q_x \quad q_y \quad q_z]^\top$$

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

Black: given variables
Red: unknowns

Camera Jacobian



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

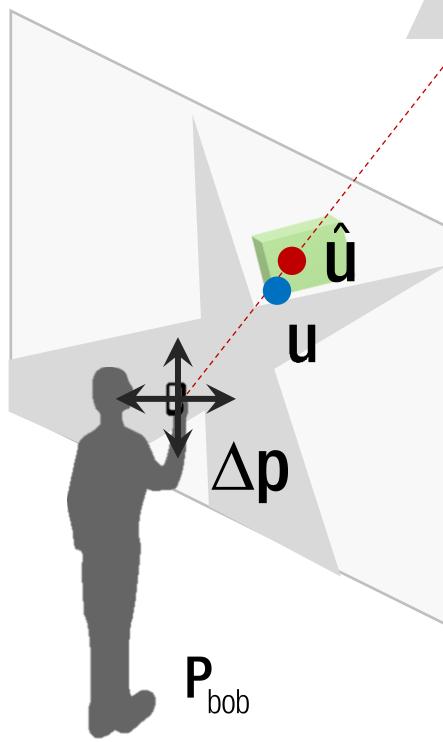
3+4 parameters

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

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

Black: given variables
Red: unknowns

Camera Jacobian



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

3+4 parameters

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

3+4 parameters

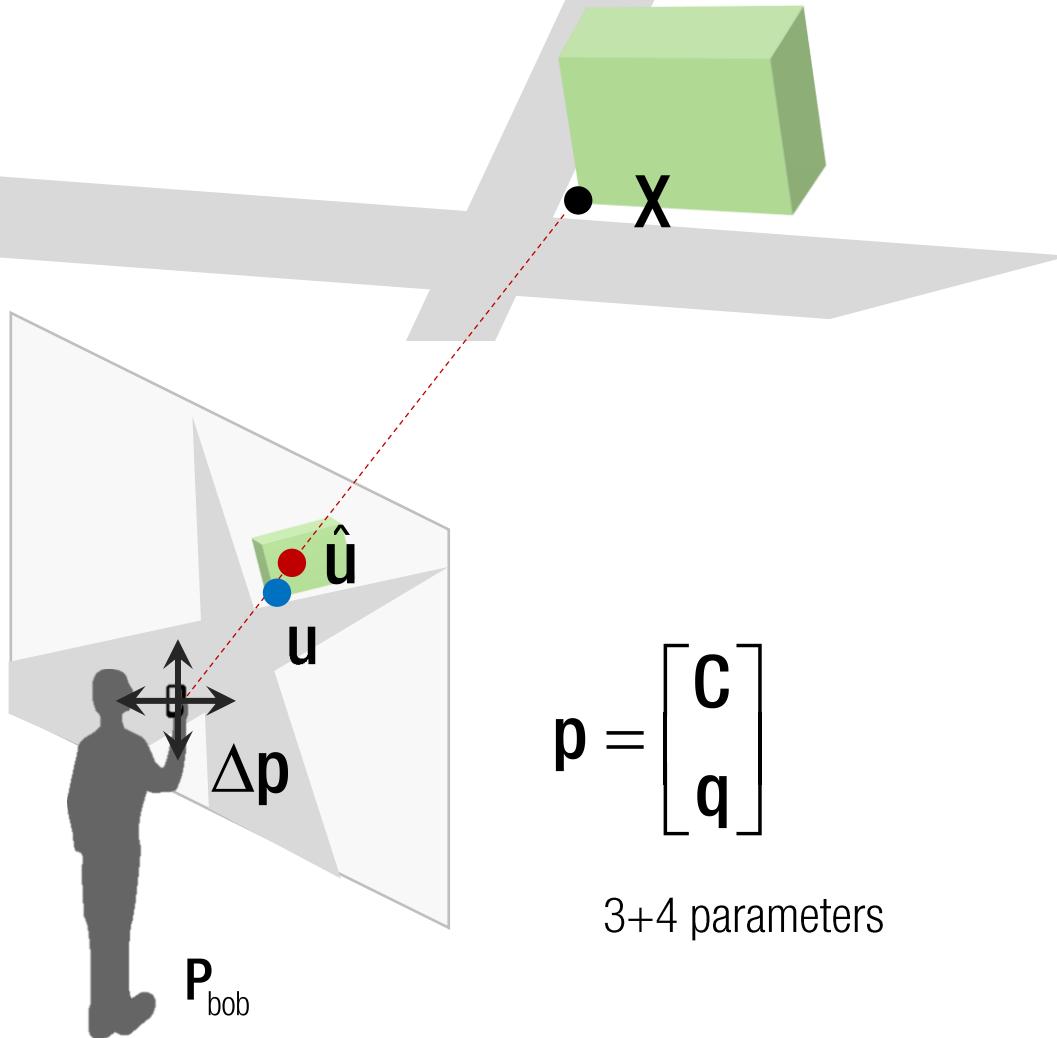
$$E_{\text{geom}} = \left(\frac{u}{w} - x \right)^2 + \left(\frac{v}{w} - y \right)^2 \quad \text{where} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{K}\mathbf{R}(\mathbf{x} - \mathbf{c})$$

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

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

Black: given variables
Red: unknowns

Camera Jacobian



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

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

$$\rightarrow \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \left[\frac{\partial}{\partial \mathbf{c}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad \frac{\partial}{\partial \mathbf{q}} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \right]$$

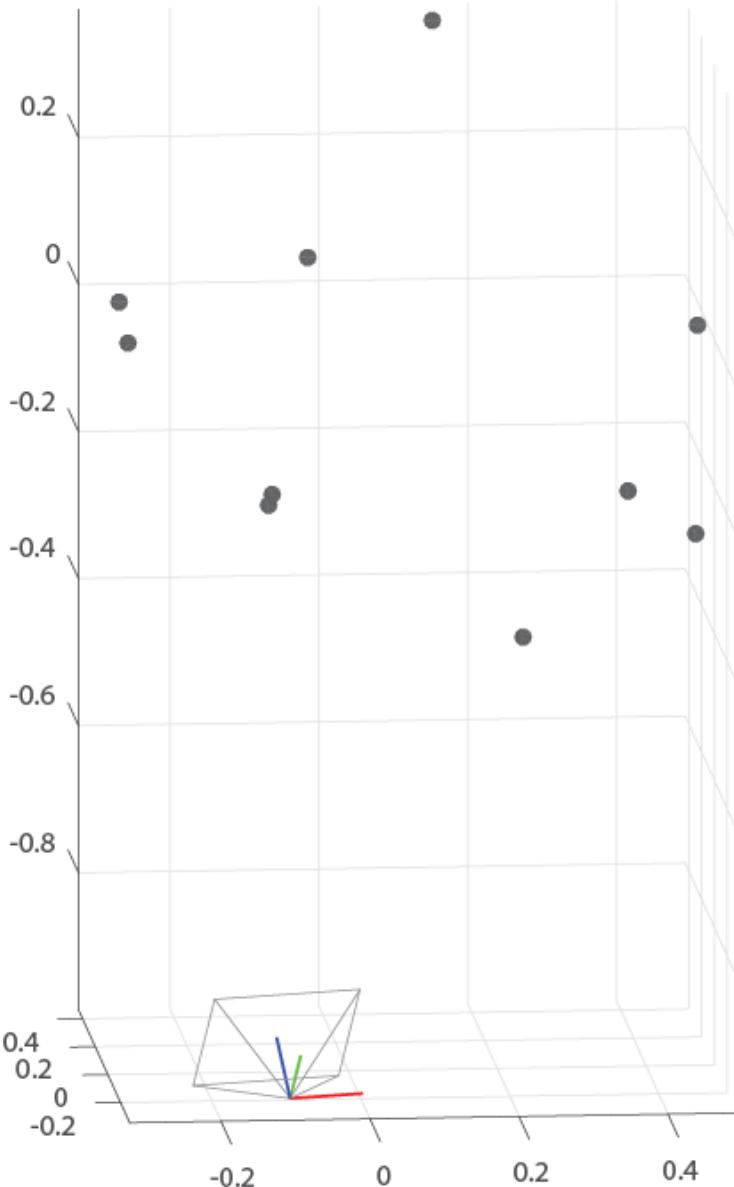
$$f(\mathbf{p}) = \begin{bmatrix} u \\ w \\ v \\ w \end{bmatrix} \rightarrow \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}} \begin{bmatrix} u \\ w \\ v \\ w \end{bmatrix} = \begin{bmatrix} \frac{w \frac{\partial u}{\partial \mathbf{p}} - u \frac{\partial w}{\partial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

Algorithm 2 Nonlinear Camera Pose Refinement

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

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

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
```

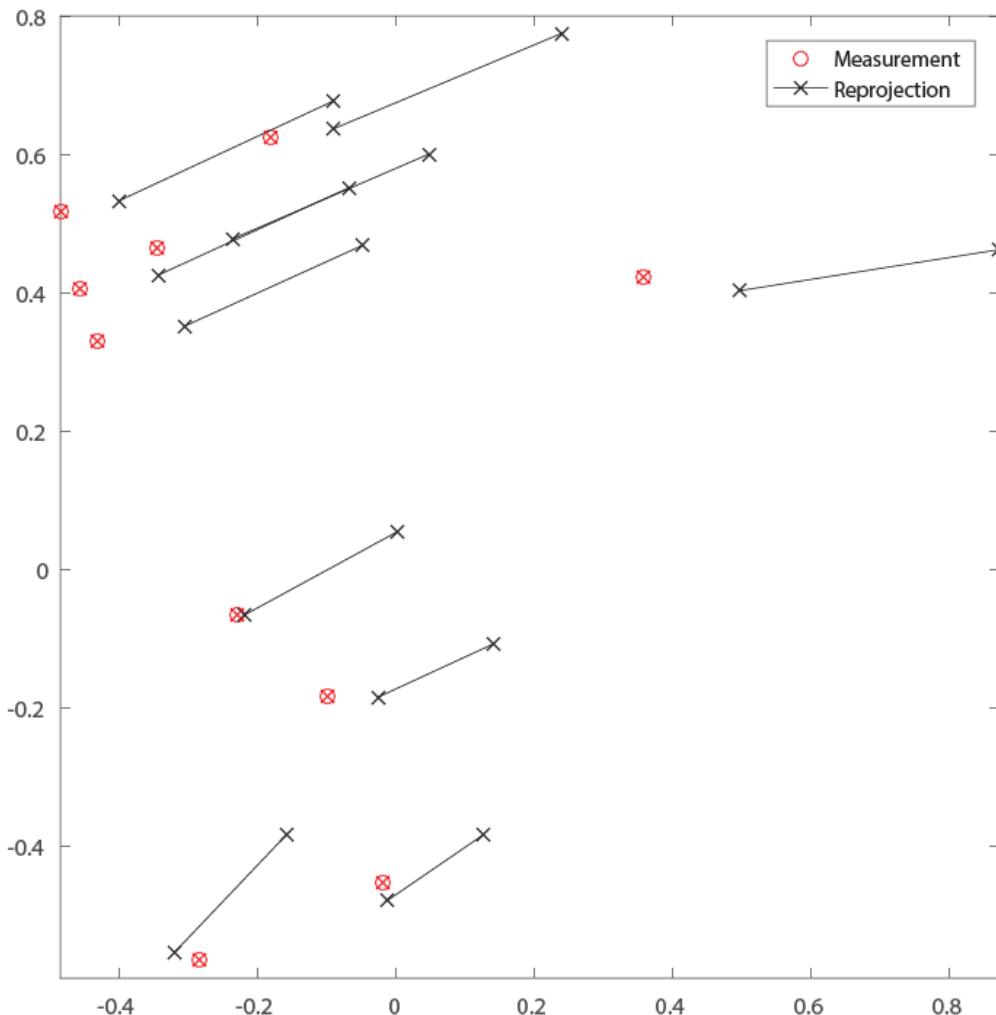
```
u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];
```

```
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
```

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

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
```

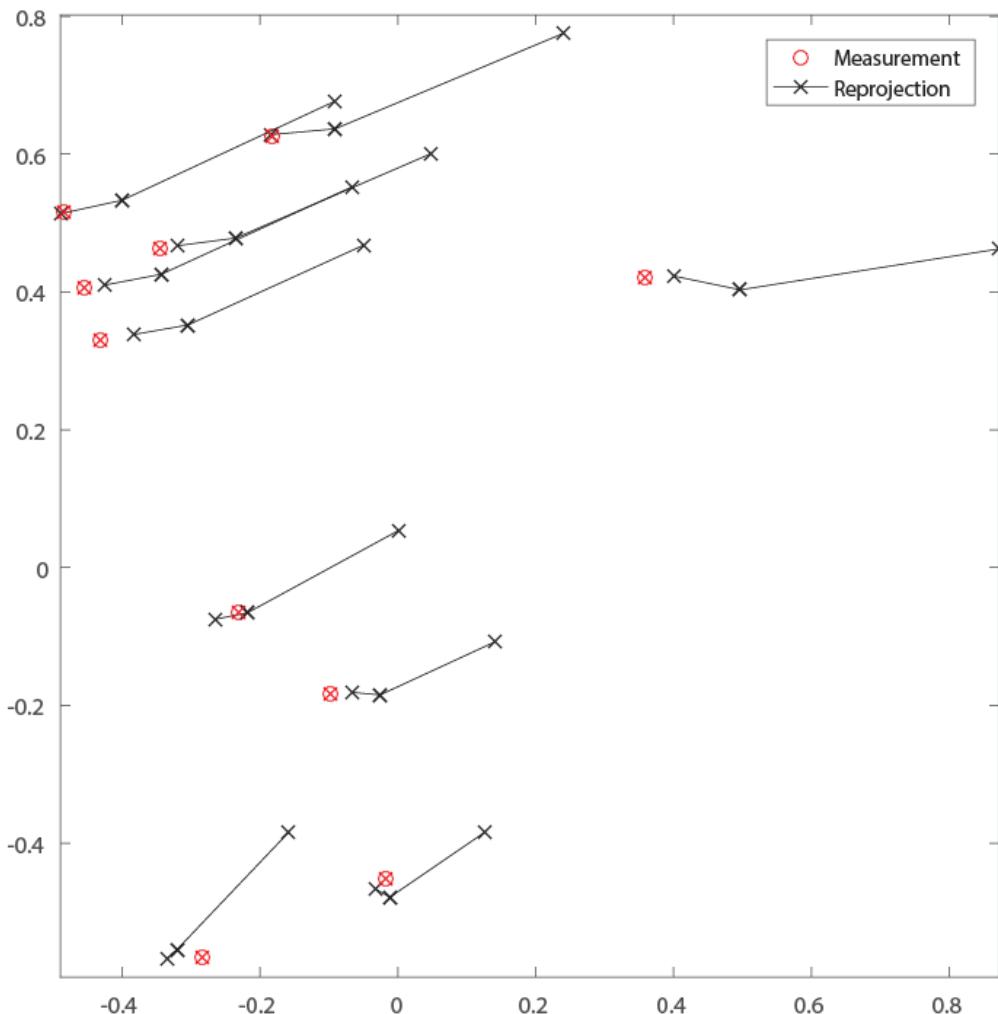
```
u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];
```

```
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
```

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

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
```

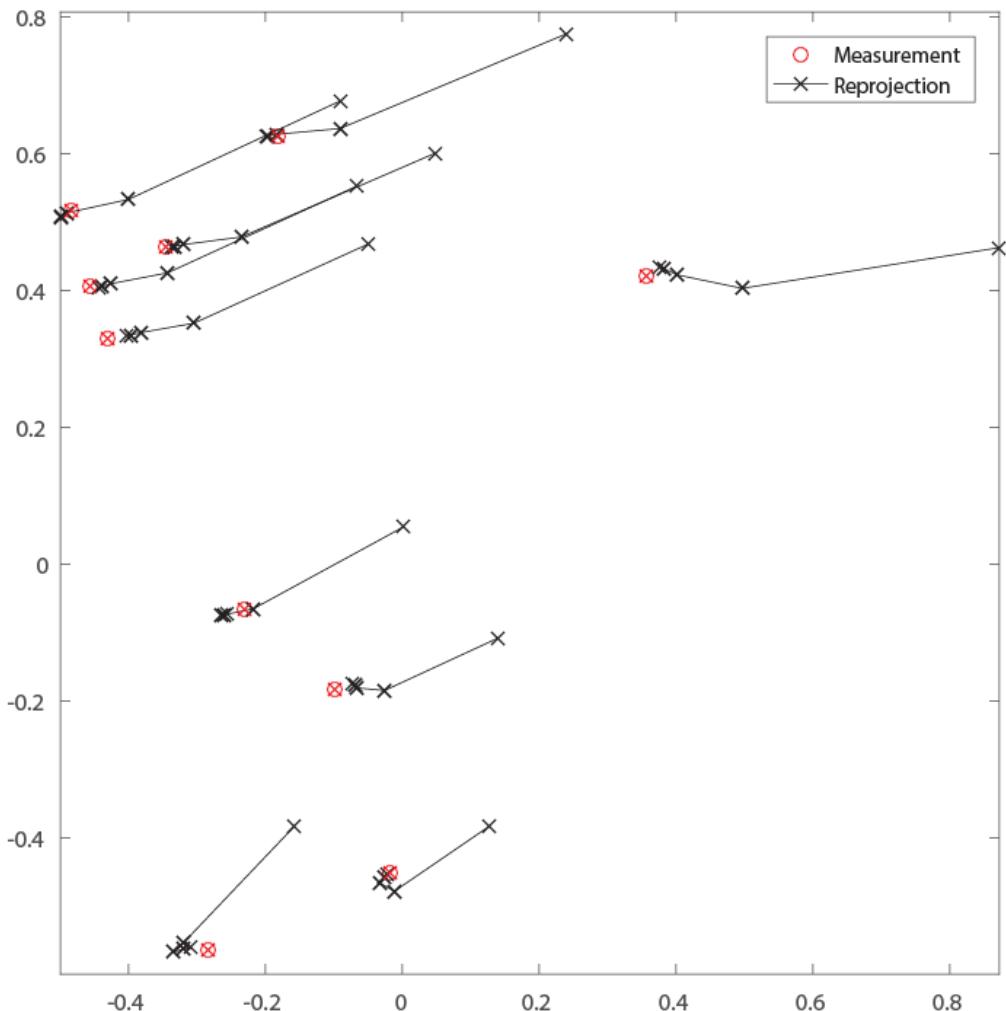
```
u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];
```

```
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
```

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

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
```

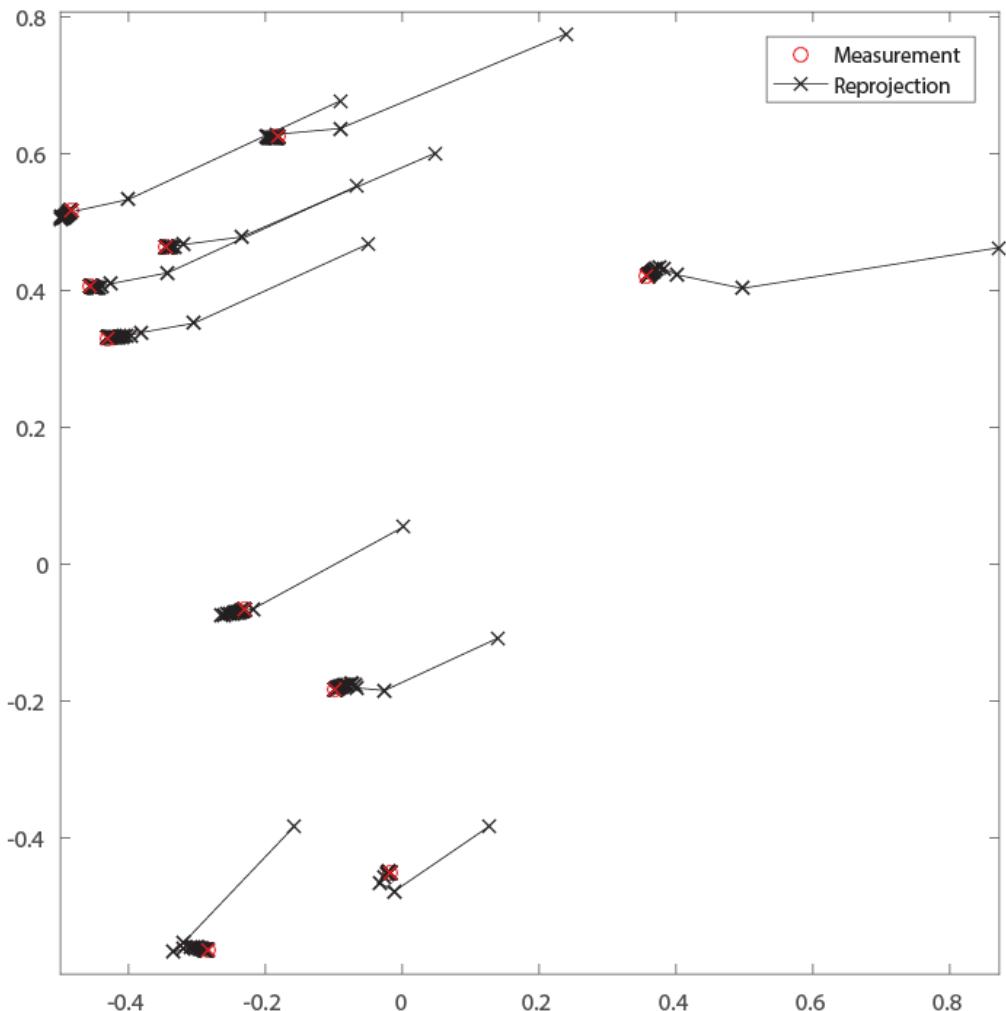
```
u1 = K*R1*[eye(3) -C1]*[X'; ones(1,nPoints)];
u1 = [u1(1,:)./u1(3,:); u1(2,:)./u1(3,:)];
```

```
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
```

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

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
```

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

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