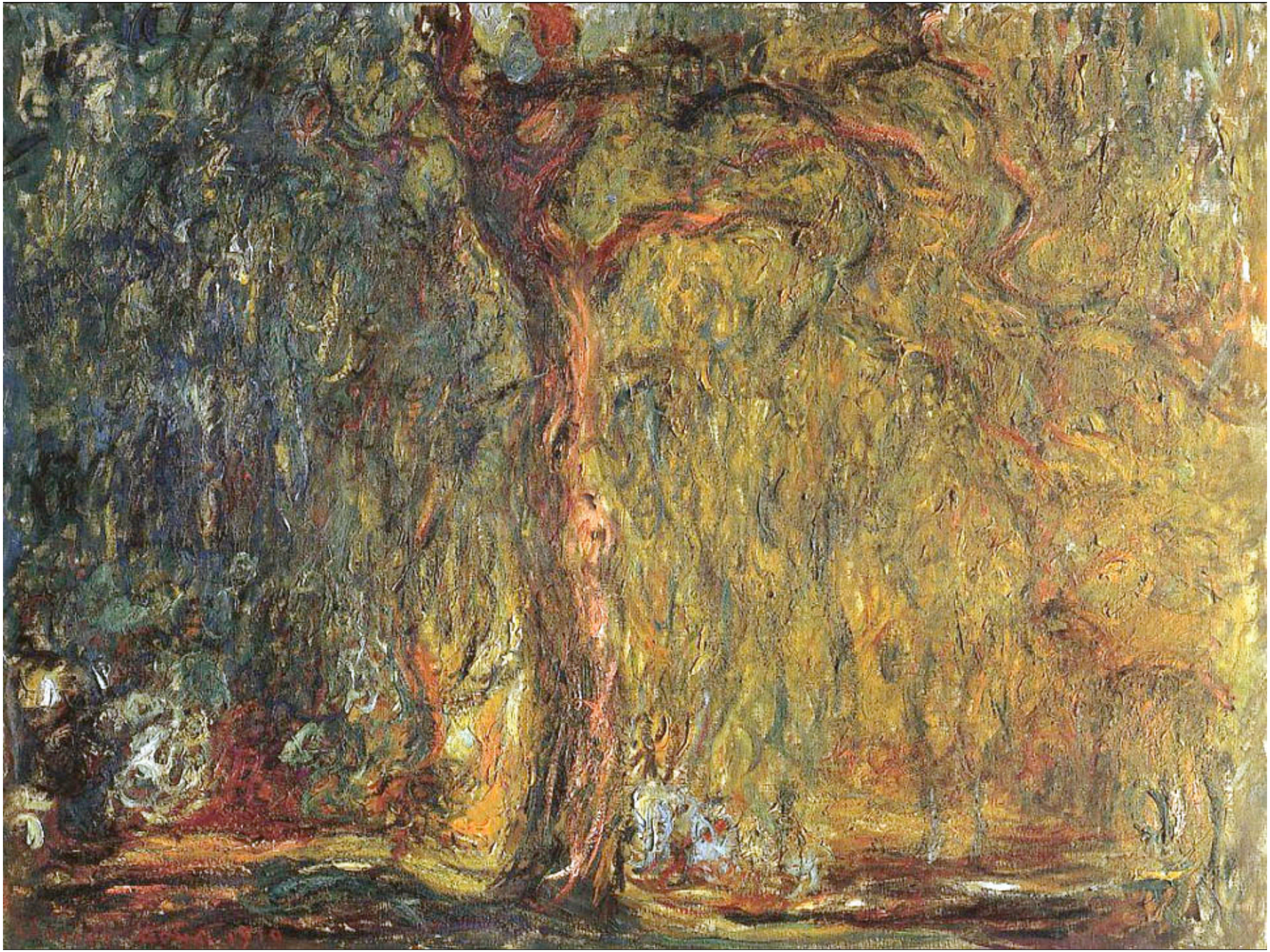


Reconnaissance d'objets et vision artificielle

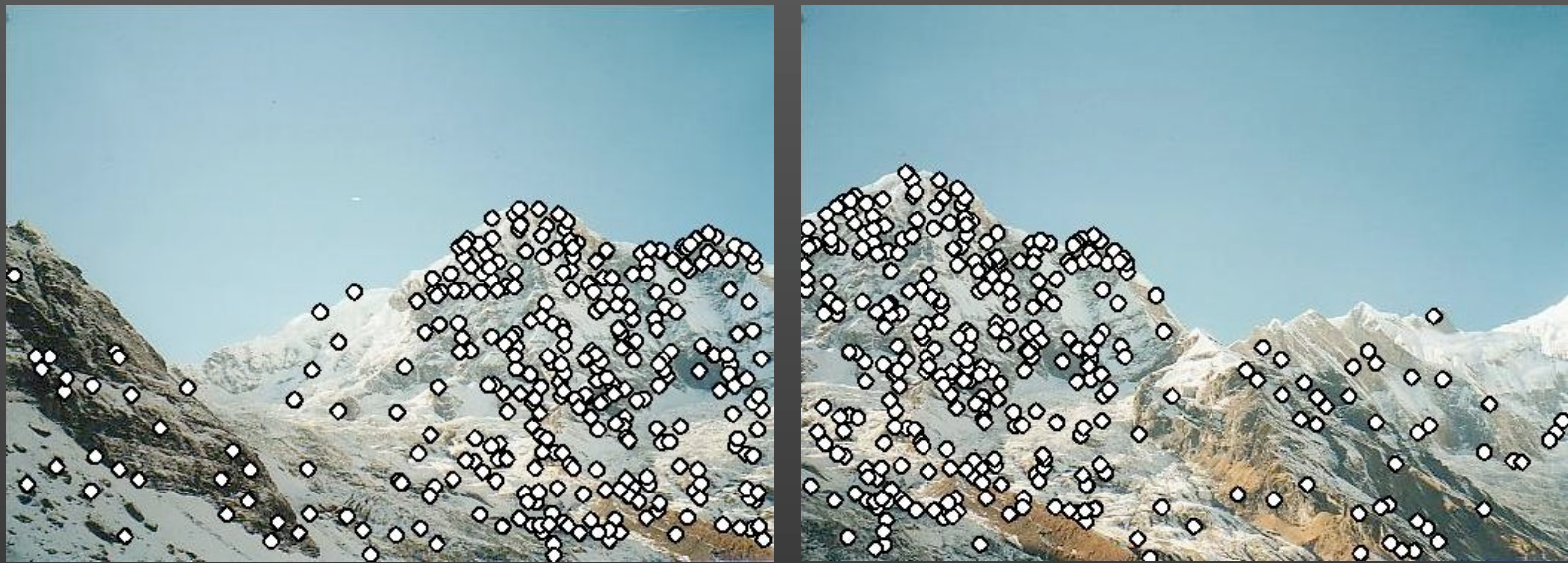
Jean Ponce (ponce@di.ens.fr)
http://www.di.ens.fr/~ponce
Equipe-projet WILLOW
ENS/INRIA/CNRS UMR 8548
Laboratoire d'Informatique
Ecole Normale Supérieure, Paris

Outline

- Motivation: Making mosaics
- Perspetive and weak perspective
- Coordinates changes
- Intrinsic and extrensic parameters
- Affine registration
- Projective registration

Feature-based alignment outline

Feature-based alignment outline



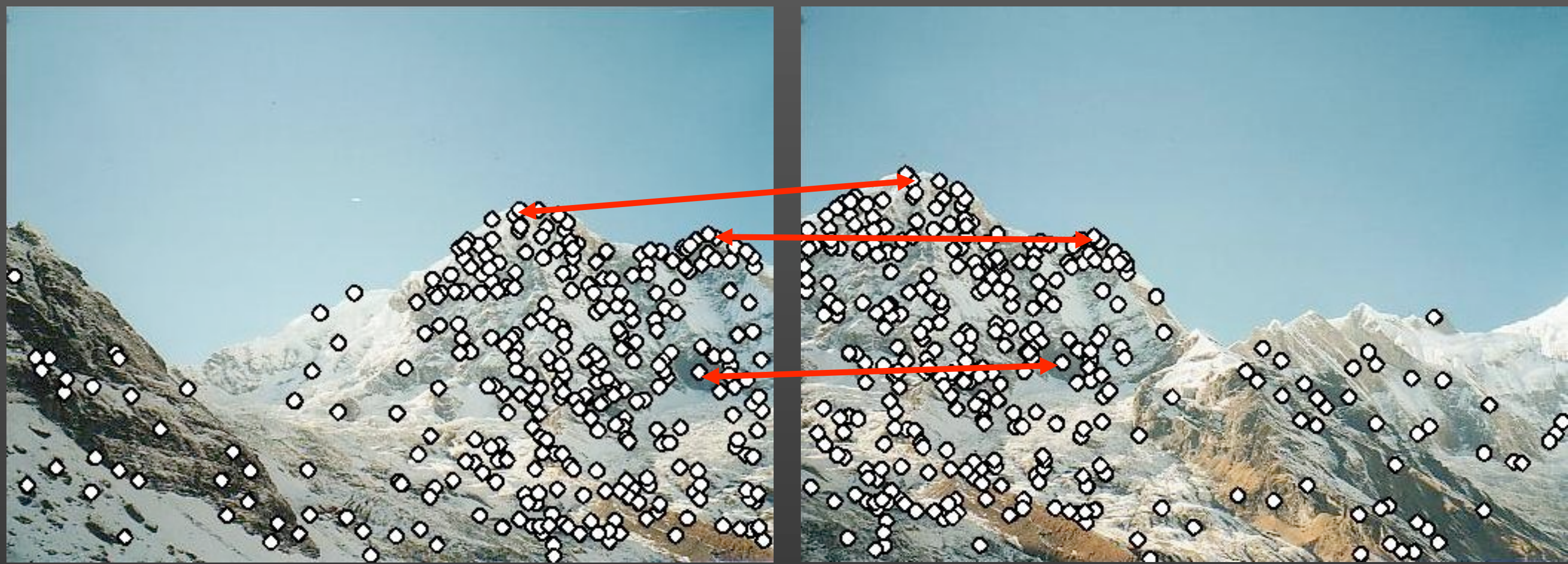
Extract features

Feature-based alignment outline

Extract features

Compute *putative matches*

Feature-based alignment outline



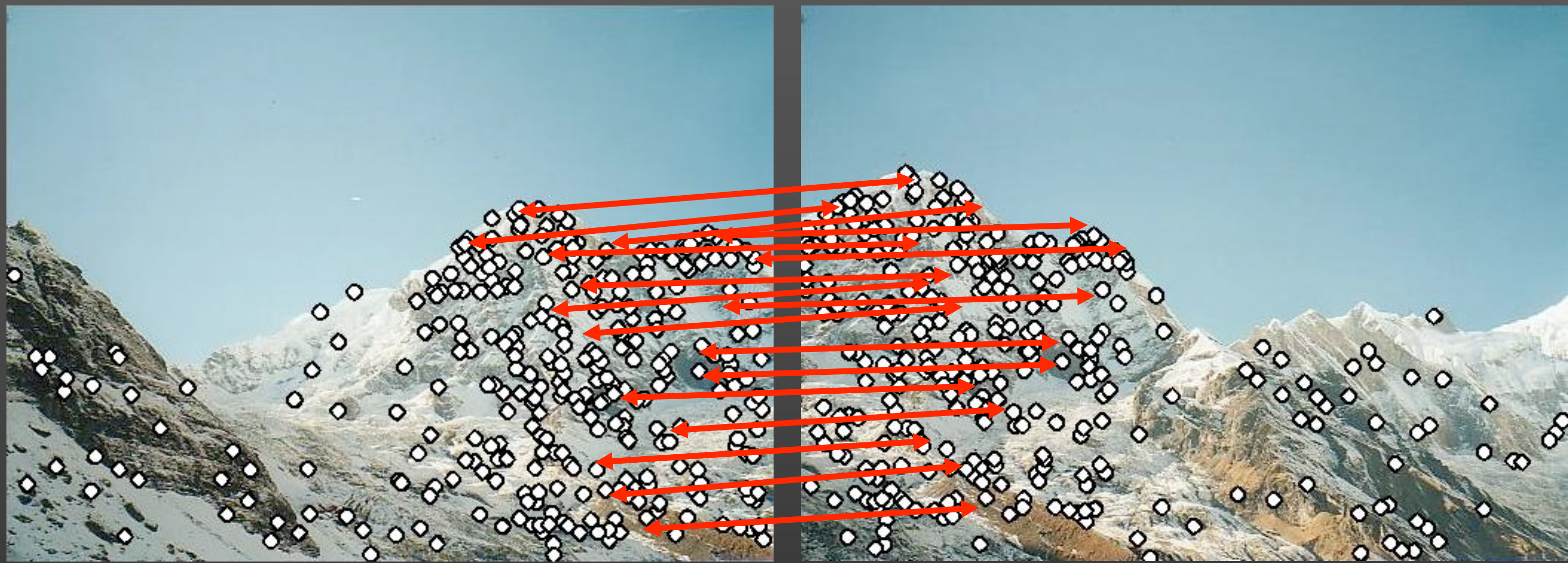
Extract features

Compute *putative matches*

Loop:

- *Hypothesize* transformation *T* (small group of putative matches that are related by *T*)

Feature-based alignment outline



Extract features

Compute *putative matches*

Loop:

- *Hypothesize* transformation *T* (small group of putative matches that are related by *T*)
- *Verify* transformation (search for other matches consistent with *T*)

Feature-based alignment outline

Extract features

Compute *putative matches*

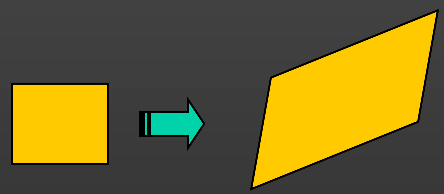
Loop:

- *Hypothesize* transformation *T* (small group of putative matches that are related by *T*)
- *Verify* transformation (search for other matches consistent with *T*)

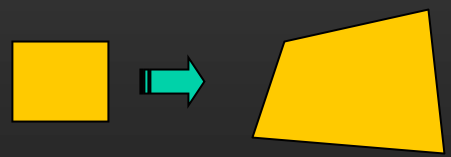
2D transformation models

Similarity
(translation,
scale, rotation)

Affine

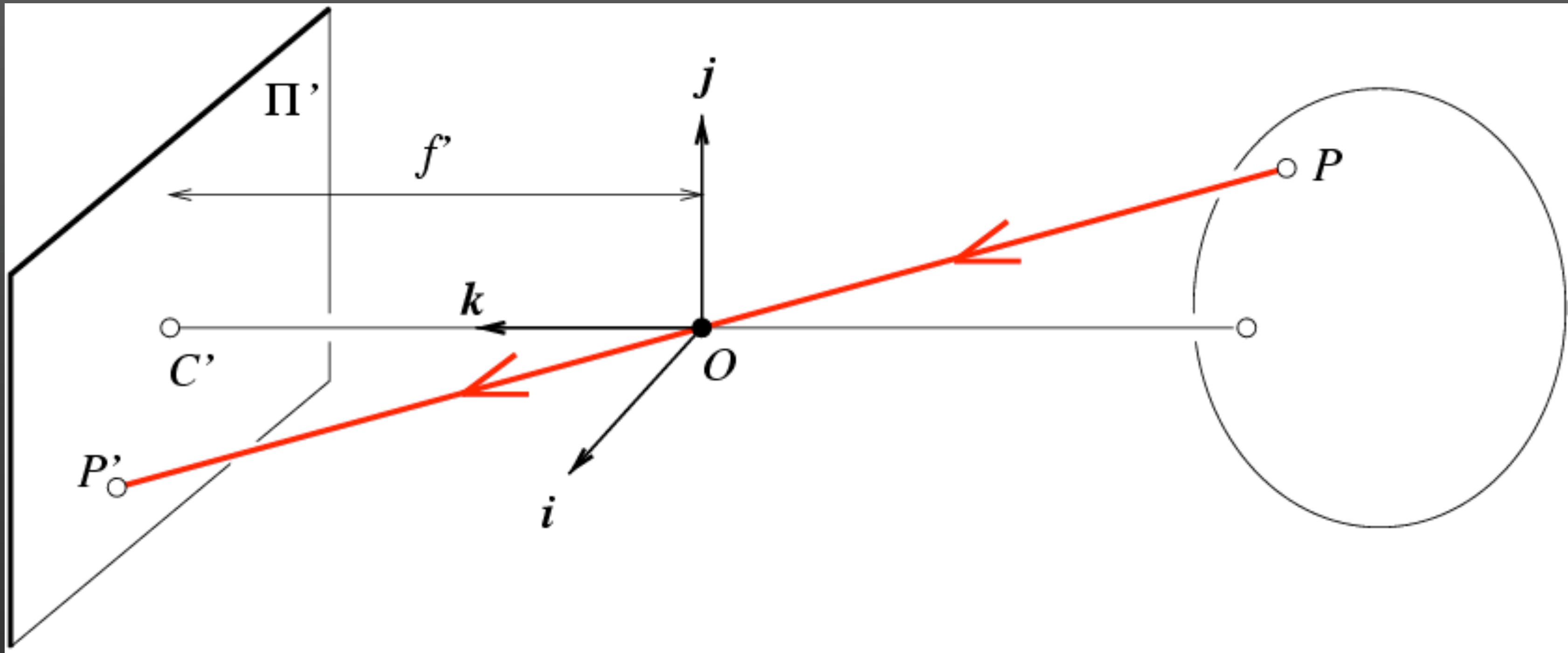


Projective
(homography)



Why these transformations ???

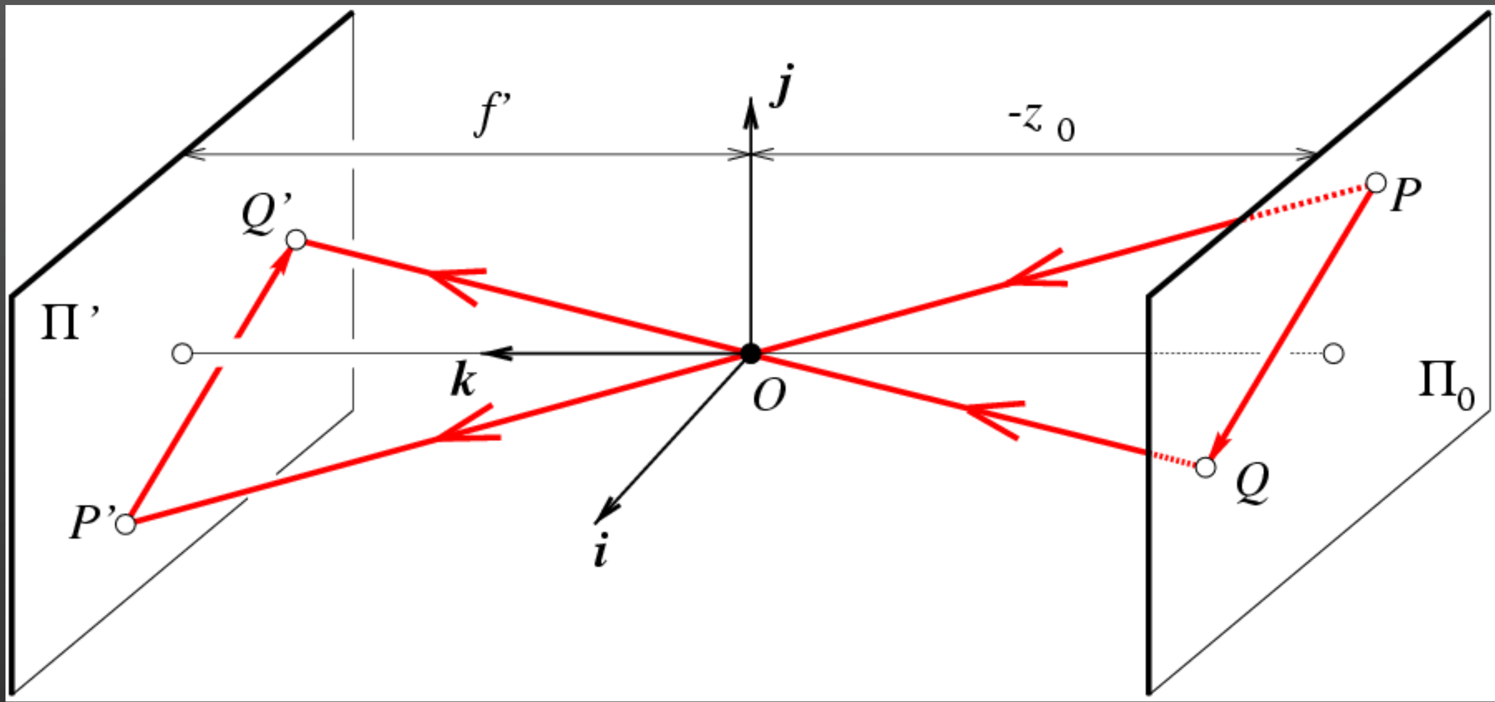
Pinhole perspective equation



$$\begin{cases} x' = f' \dfrac{x}{z} \\ y' = f' \dfrac{y}{z} \end{cases}$$

NOTE: *z* is always negative..

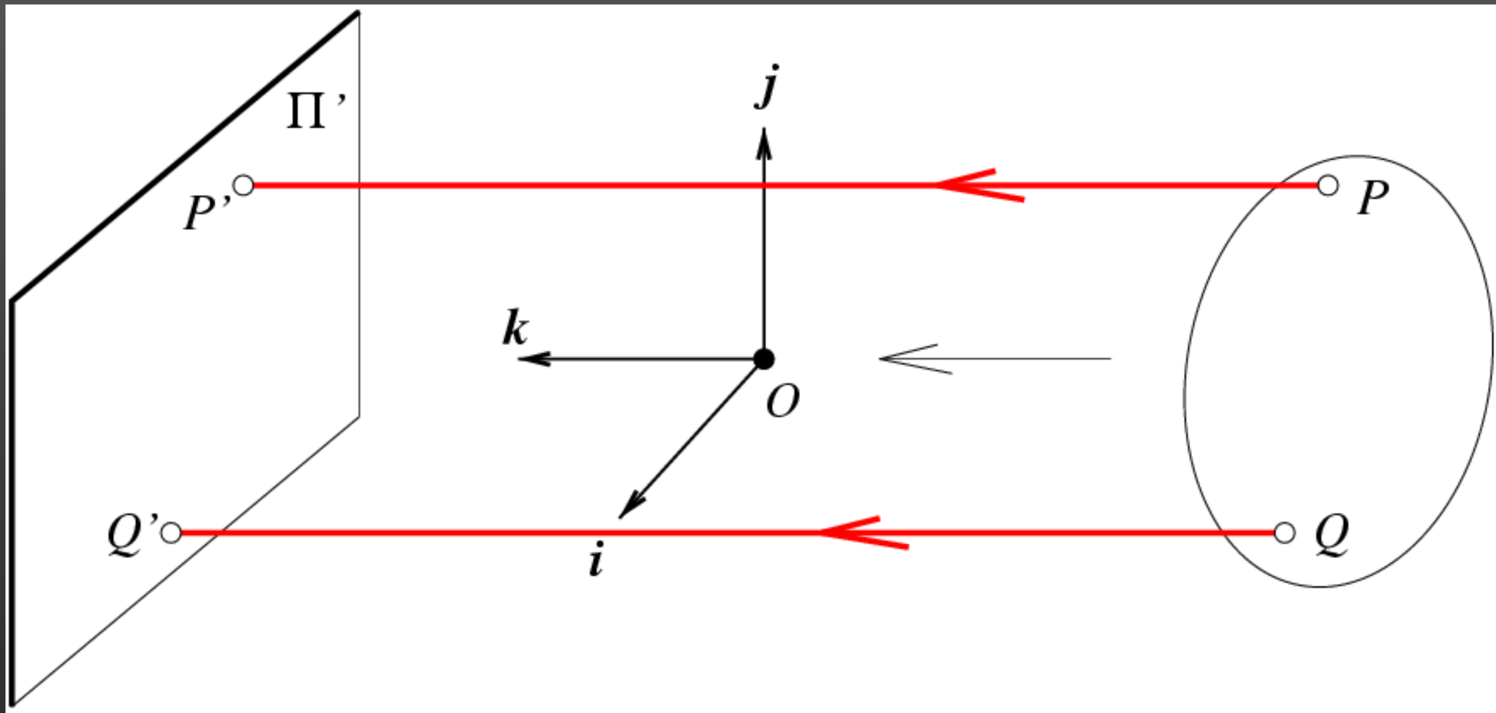
Affine models: Weak perspective projection



$$\begin{cases} x' = -mx \\ y' = -my \end{cases} \quad \text{where} \quad m = -\frac{f'}{z_0}$$ is the magnification.

When the scene relief is small compared its distance from the Camera, *m* can be taken constant: weak perspective projection.

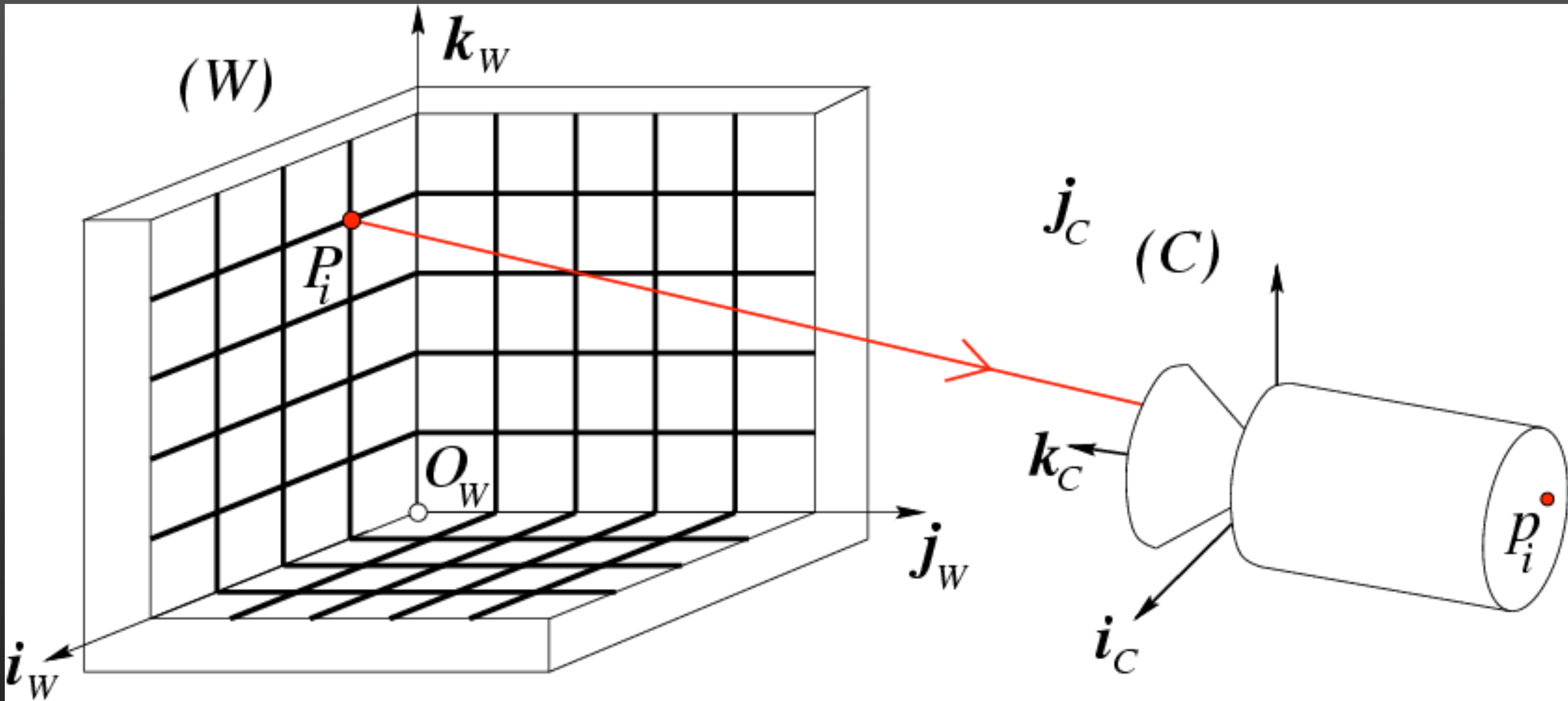
Affine models: Orthographic projection



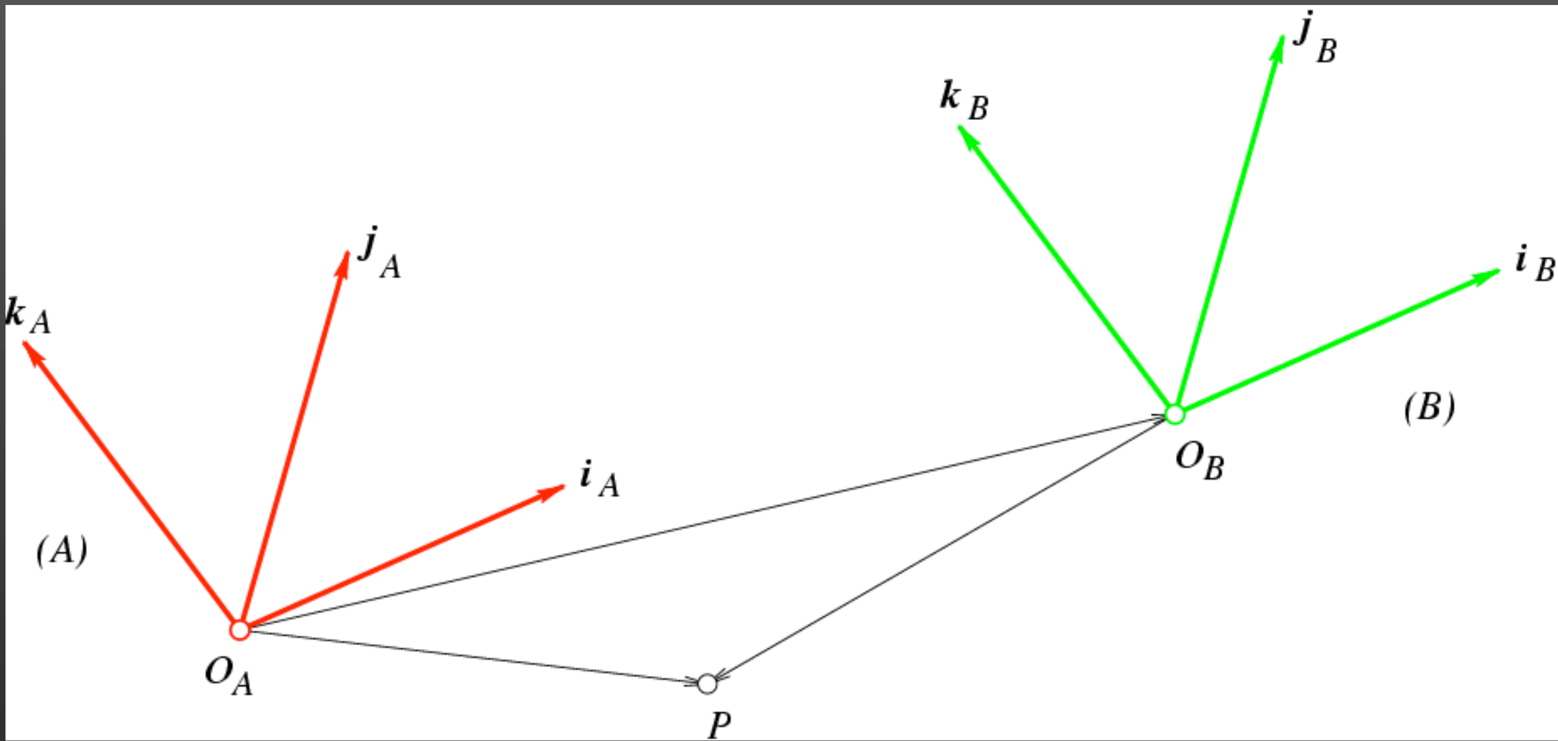
$$\begin{cases} x' = x \\ y' = y \end{cases}$$

When the camera is at a (roughly constant) distance from the scene, take m=1.

Analytical camera geometry

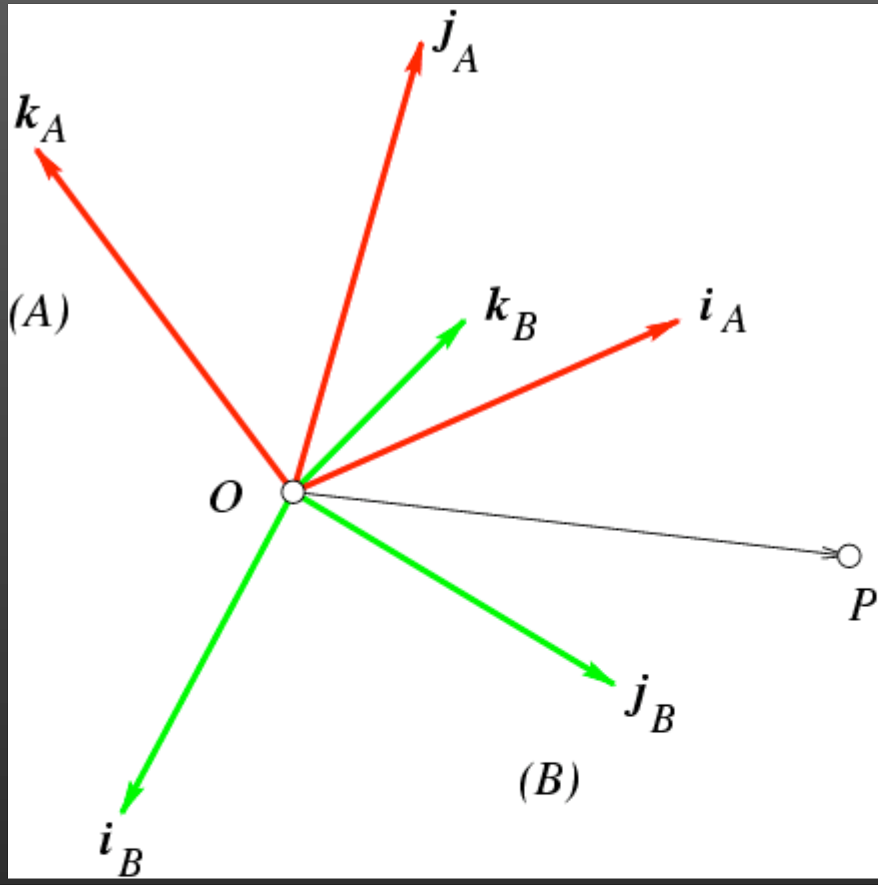


Coordinate Changes: Pure Translations



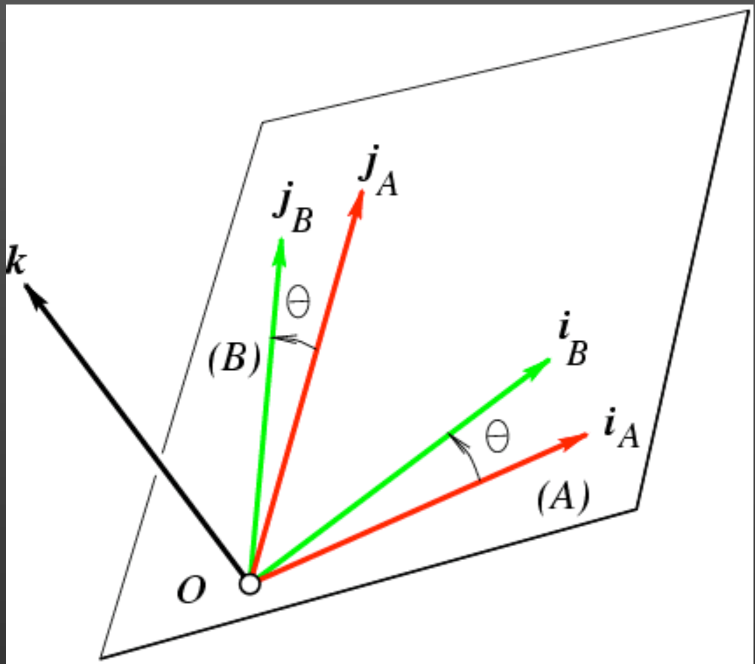
$$\overrightarrow{O_BP} = \overrightarrow{O_BO_A} + \overrightarrow{O_AP} \quad , \quad {}^BP = {}^AP + {}^BO_A$$

Coordinate Changes: Pure Rotations

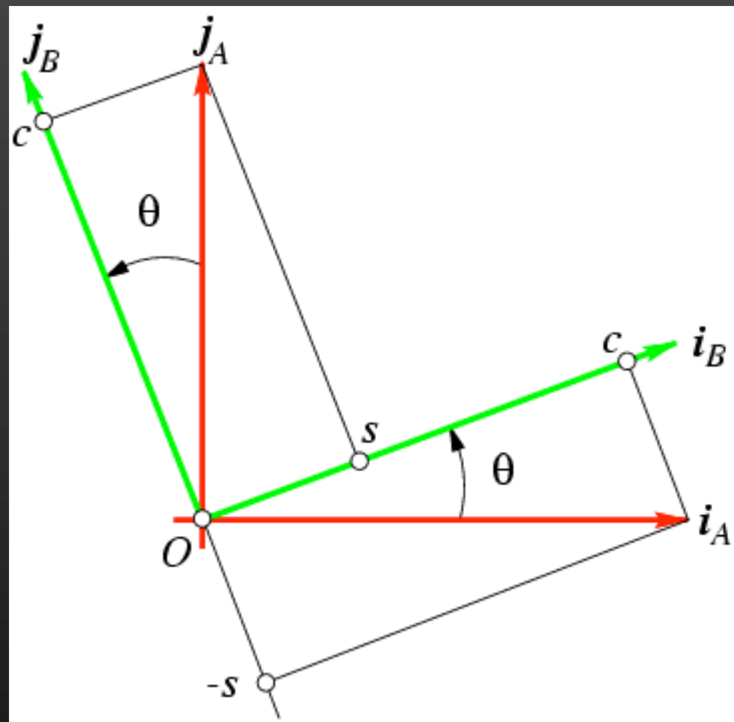


$$ {}^B_A R = \begin{bmatrix} \mathbf{i}_A.\mathbf{i}_B & \mathbf{j}_A.\mathbf{i}_B & \mathbf{k}_A.\mathbf{i}_B \\ \mathbf{i}_A.\mathbf{j}_B & \mathbf{j}_A.\mathbf{j}_B & \mathbf{k}_A.\mathbf{j}_B \\ \mathbf{i}_A.\mathbf{k}_B & \mathbf{j}_A.\mathbf{k}_B & \mathbf{k}_A.\mathbf{k}_B \end{bmatrix} = \begin{bmatrix} {}^A\mathbf{i}_B^T \\ {}^A\mathbf{j}_B^T \\ {}^A\mathbf{k}_B^T \end{bmatrix} = \begin{bmatrix} {}^B\mathbf{i}_A & {}^B\mathbf{j}_A & {}^B\mathbf{k}_A \end{bmatrix} $$

Coordinate Changes: Rotations about the z Axis



$$
{}^B_A R = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$



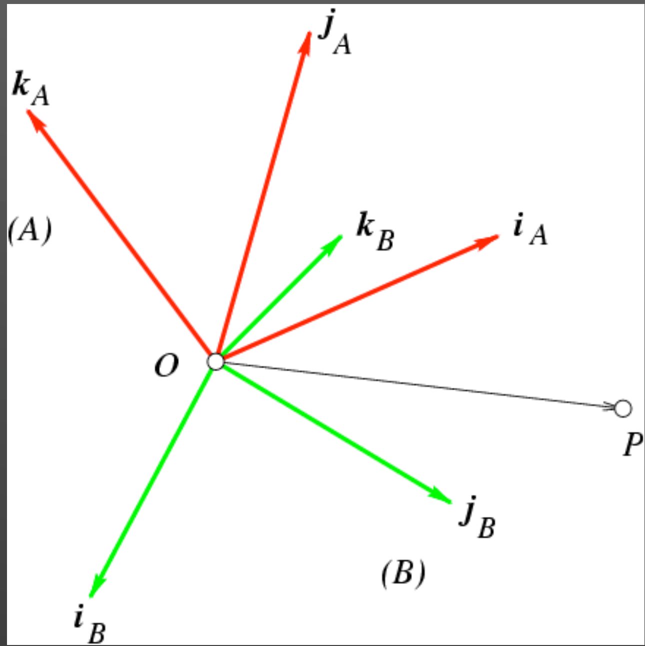
A rotation matrix is characterized by the following properties:

- Its inverse is equal to its transpose, and
- its determinant is equal to 1.

Or equivalently:

- Its rows (or columns) form a right-handed orthonormal coordinate system.

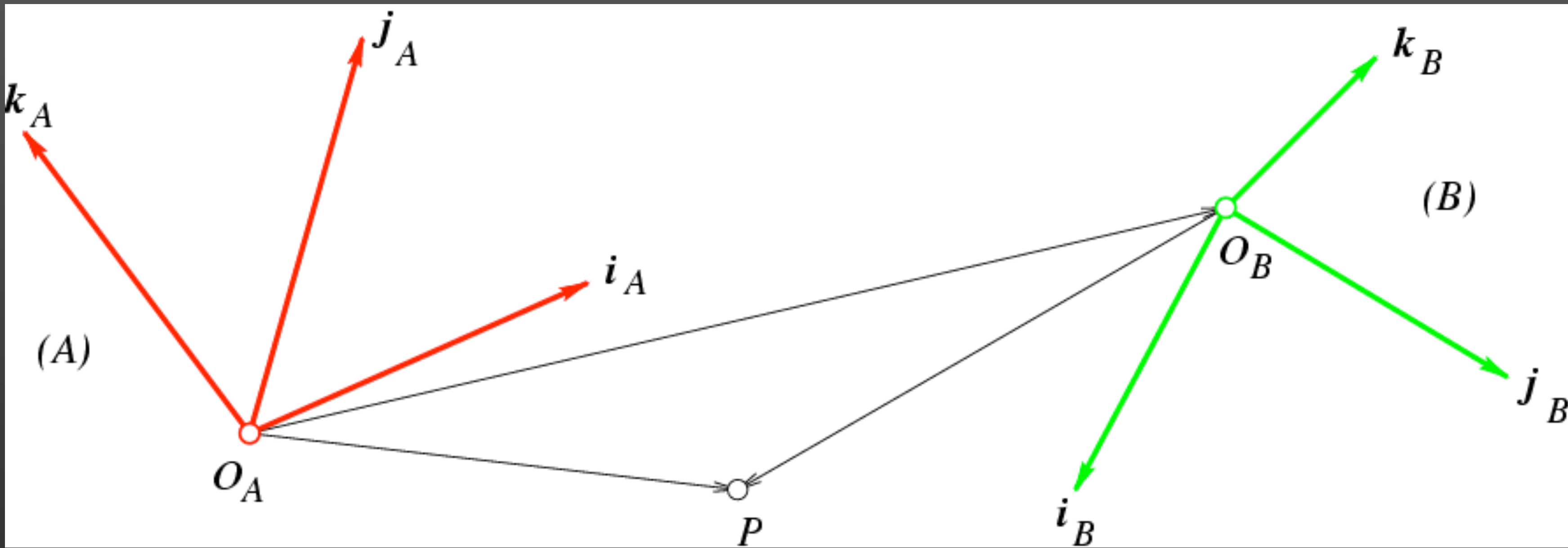
Coordinate changes: pure rotations



$$\overrightarrow{OP} = \begin{bmatrix} \mathbf{i}_A & \mathbf{j}_A & \mathbf{k}_A \end{bmatrix} \begin{bmatrix} {}^A x \\ {}^A y \\ {}^A z \end{bmatrix} = \begin{bmatrix} \mathbf{i}_B & \mathbf{j}_B & \mathbf{k}_B \end{bmatrix} \begin{bmatrix} {}^B x \\ {}^B y \\ {}^B z \end{bmatrix}$$

$$\Rightarrow \quad {}^B P = {}^B_A R \, {}^A P$$

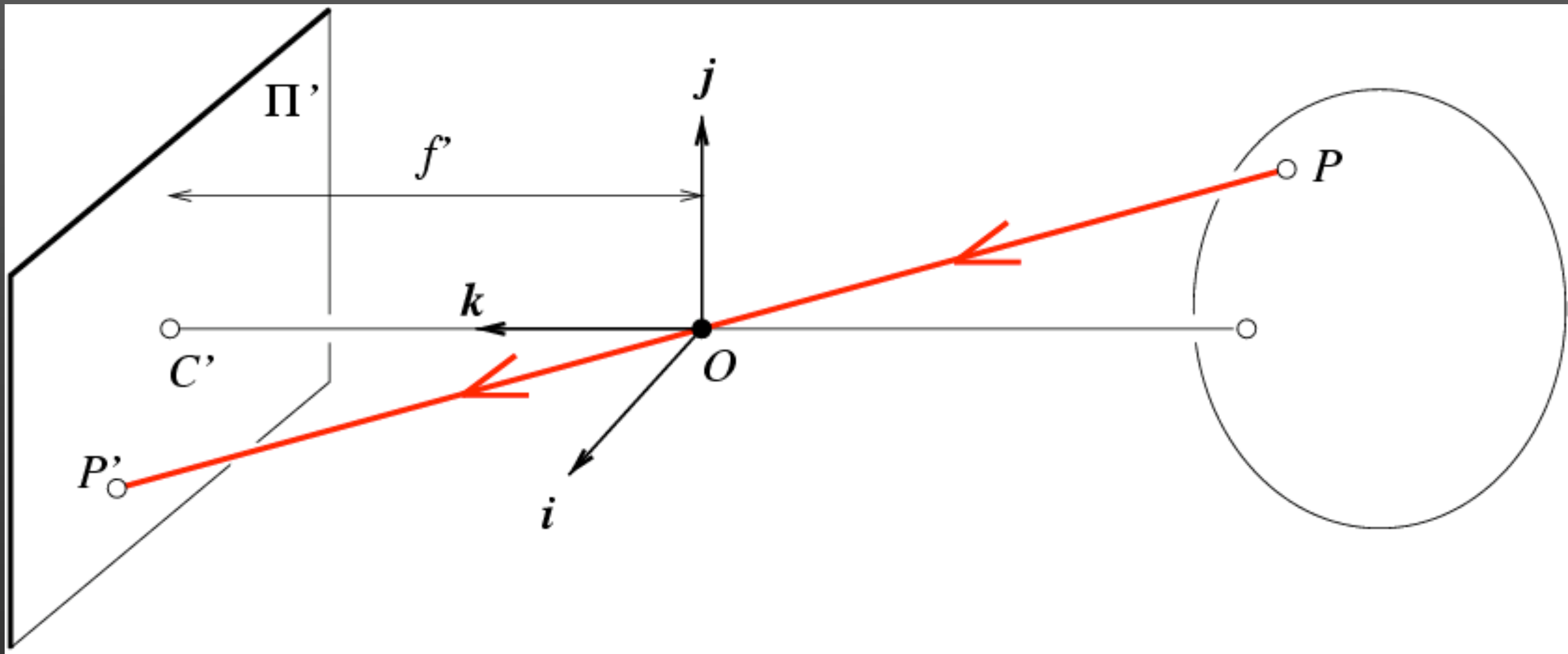
Coordinate Changes: Rigid Transformations



$$\begin{bmatrix} {}^{B}P \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{B}_{A}R & {}^{B}O_A \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} {}^{A}P \\ 1 \end{bmatrix}$$

$${}^{B}P = {}^{B}_{A}R\,{}^{A}P + {}^{B}O_A$$

Pinhole perspective equation



$$\begin{cases} x' = f'\dfrac{x}{z} \\ y' = f'\dfrac{y}{z} \end{cases}$$

NOTE: z is always negative..

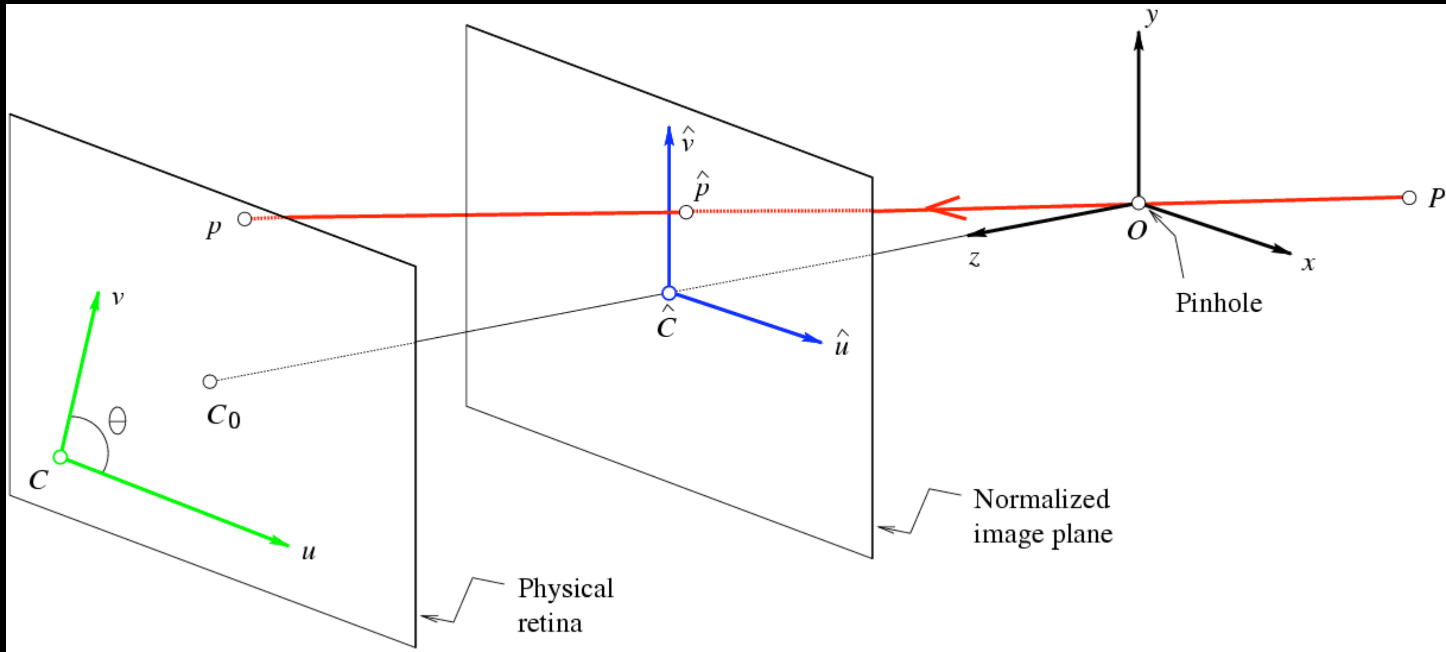
The intrinsic parameters of a camera

Units:

k,l : pixel/m

f : m

α,β : pixel



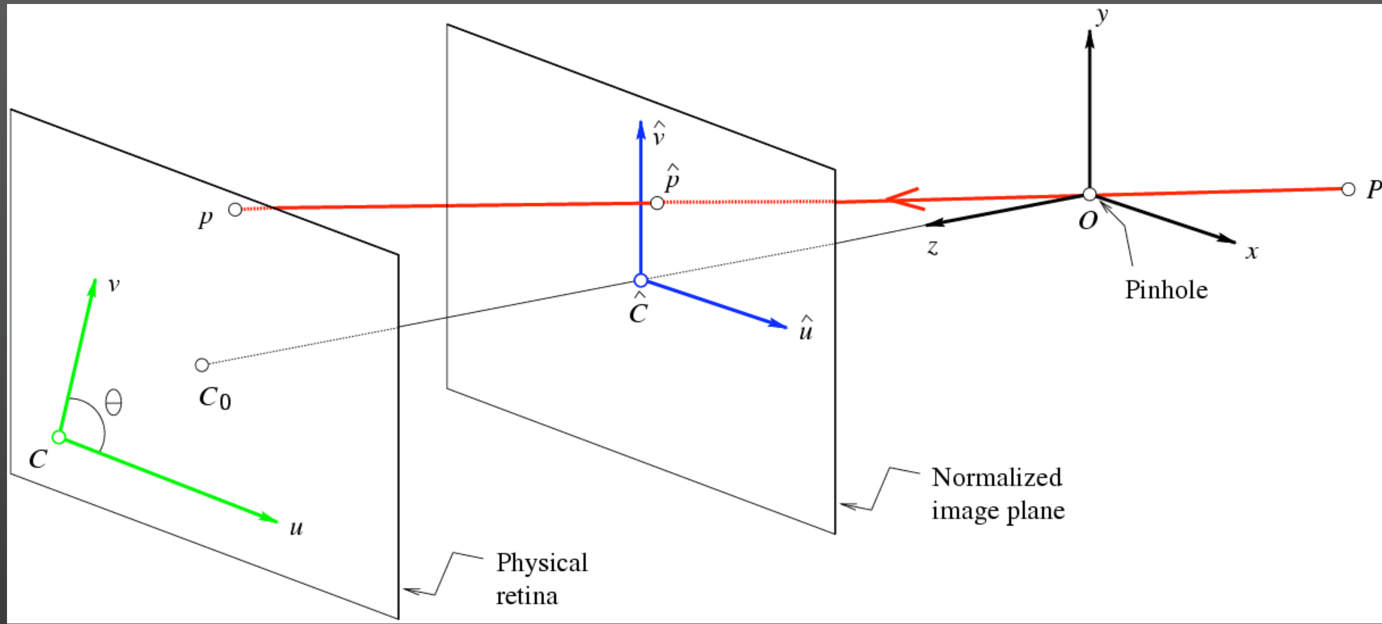
$$\begin{cases} \hat{u} = \dfrac{x}{z} \\ \hat{v} = \dfrac{y}{z} \end{cases} \iff \hat{\boldsymbol{p}} = \frac{1}{z}\begin{pmatrix} \mathrm{Id} & \mathbf{0} \end{pmatrix}\begin{pmatrix} \boldsymbol{P} \\ 1 \end{pmatrix}$$

Normalized image coordinates

Physical image coordinates

$$\begin{cases} u = kf\dfrac{x}{z} \\ v = lf\dfrac{y}{z} \end{cases}$$

The intrinsic parameters of a camera



Calibration matrix

$$\boldsymbol{p} = \mathcal{K}\hat{\boldsymbol{p}}, \quad \text{where} \quad \boldsymbol{p} = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \quad \text{and} \quad \mathcal{K} \stackrel{\text{def}}{=} \begin{pmatrix} \alpha & -\alpha \cot\theta & u_0 \\ 0 & \dfrac{\beta}{\sin\theta} & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

The perspective projection equation

$$\boldsymbol{p} = \frac{1}{z}\mathcal{M}\boldsymbol{P}, \quad \text{where} \quad \mathcal{M} \stackrel{\text{def}}{=} (\mathcal{K} \quad \boldsymbol{0})$$

The extrinsic parameters of a camera

- When the camera frame (C) is different from the world frame (W),

$$\begin{pmatrix} {}^{C}P \\ 1 \end{pmatrix} = \begin{pmatrix} {}^{C}_{W}\mathcal{R} & {}^{C}O_W \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{pmatrix} {}^{W}P \\ 1 \end{pmatrix}.$$

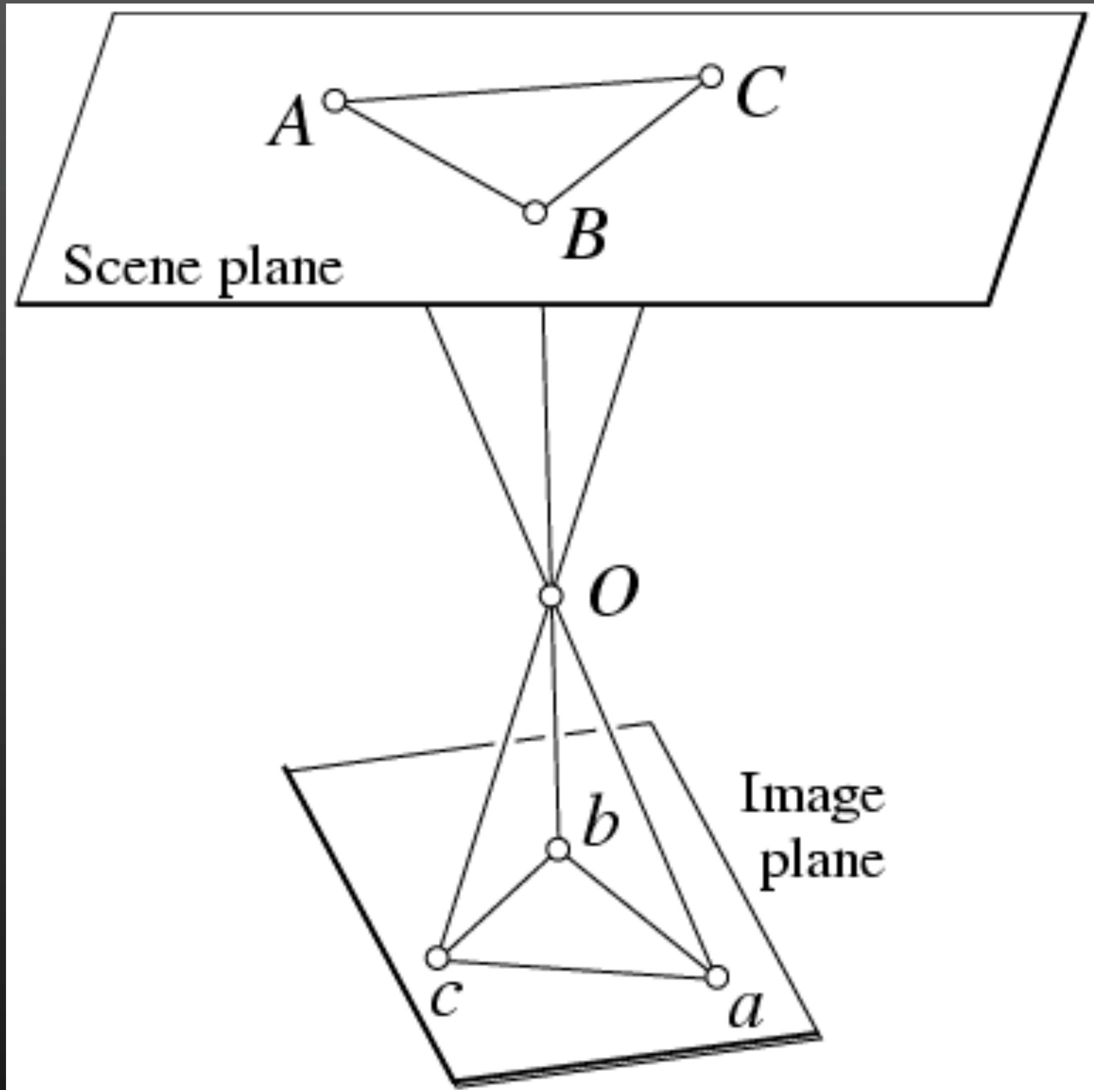
- Thus,

$$\boxed{\boldsymbol{p} = \frac{1}{z}\mathcal{M}\boldsymbol{P},} \quad \text{where} \quad \begin{cases} \mathcal{M} = \mathcal{K}\begin{pmatrix} \mathcal{R} & \boldsymbol{t} \end{pmatrix}, \\ \mathcal{R} = {}^{C}_{W}\mathcal{R}, \\ \boldsymbol{t} = {}^{C}O_W, \\ \boldsymbol{P} = \begin{pmatrix} {}^{W}P \\ 1 \end{pmatrix}. \end{cases}$$

- Note: z is *not* independent of $\mathcal{M}$ and $\boldsymbol{P}$:

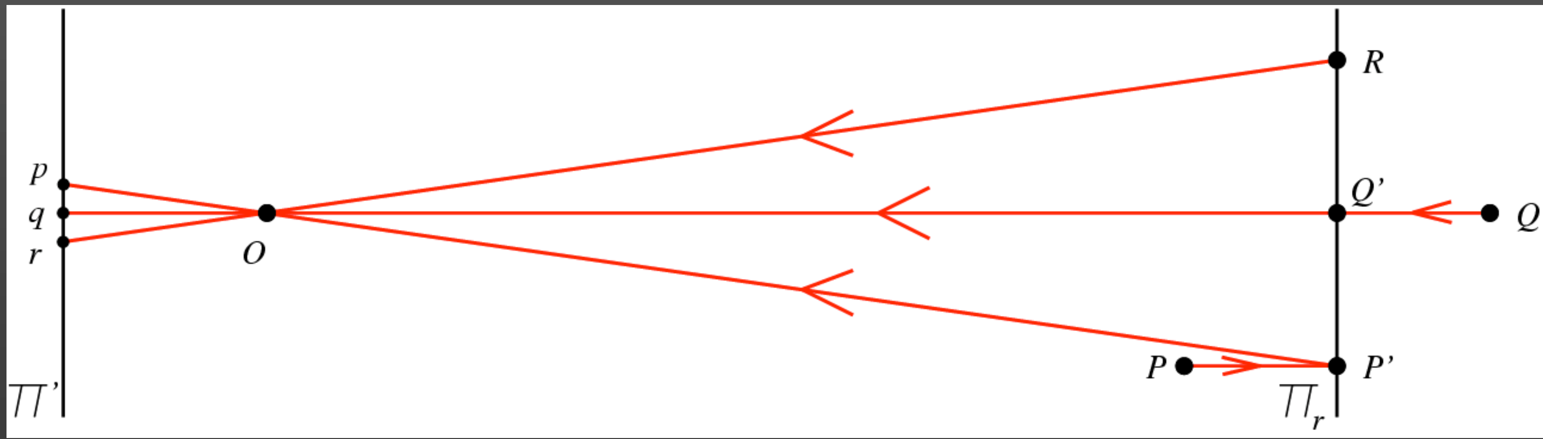
$$\mathcal{M} = \begin{pmatrix} \boldsymbol{m}_1^T \\ \boldsymbol{m}_2^T \\ \boldsymbol{m}_3^T \end{pmatrix} \Longrightarrow z = \boldsymbol{m}_3 \cdot \boldsymbol{P}, \quad \text{or} \quad \begin{cases} u = \dfrac{\boldsymbol{m}_1 \cdot \boldsymbol{P}}{\boldsymbol{m}_3 \cdot \boldsymbol{P}}, \\ v = \dfrac{\boldsymbol{m}_2 \cdot \boldsymbol{P}}{\boldsymbol{m}_3 \cdot \boldsymbol{P}}. \end{cases}$$

Perspective projections induce projective transformations between planes

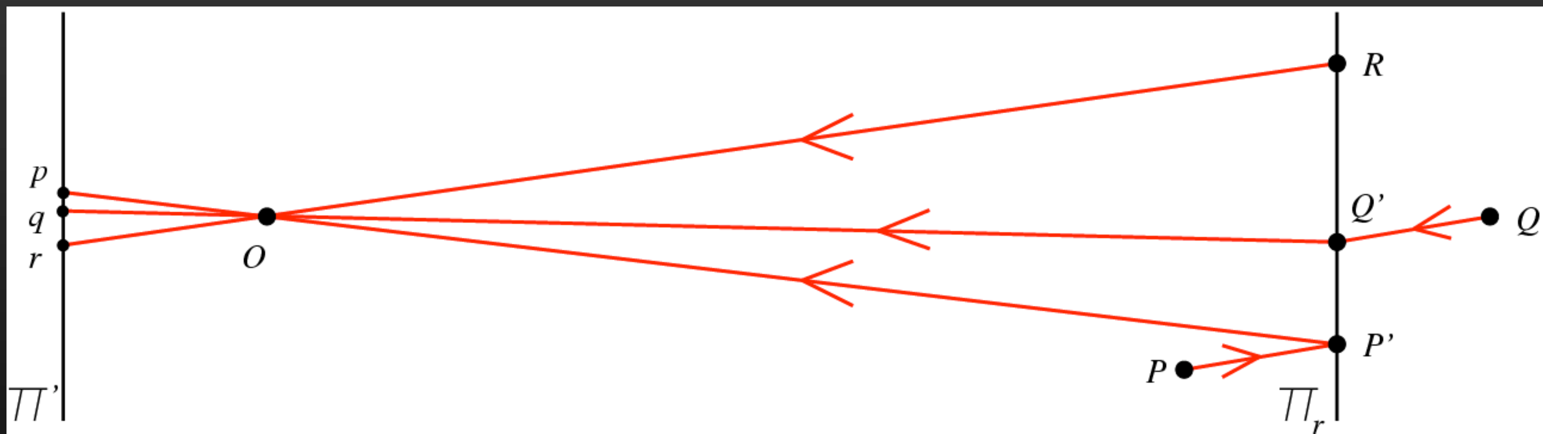


Affine cameras

Weak-perspective projection

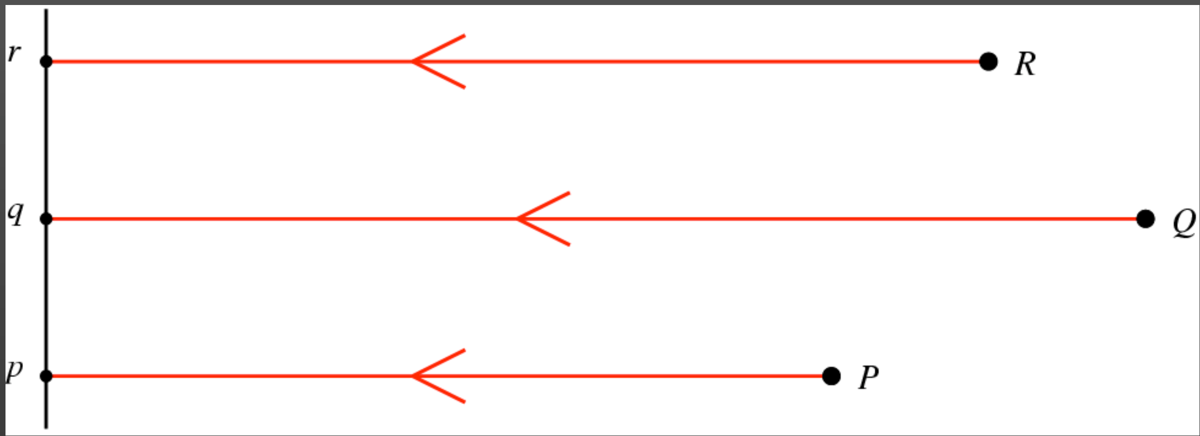


Paraperspective projection

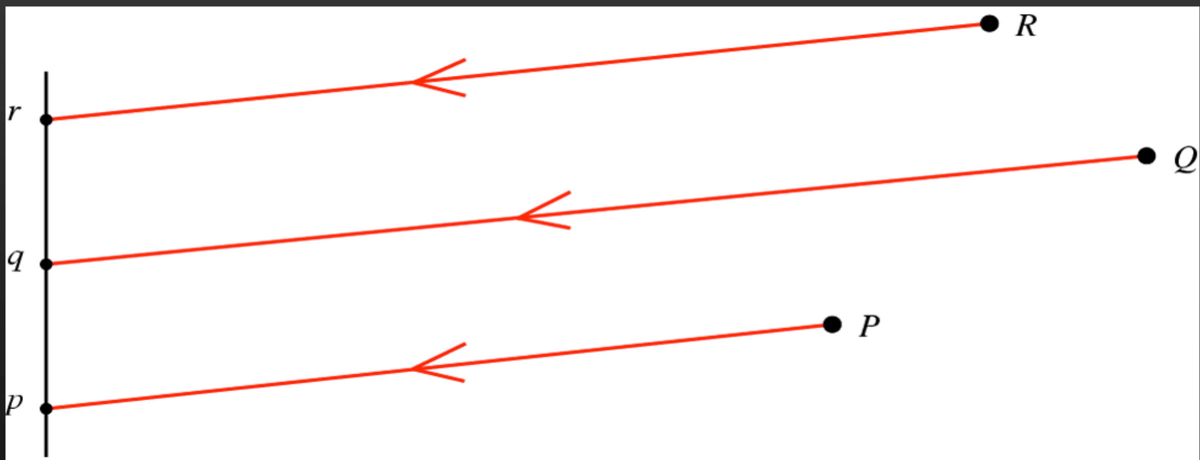


More affine cameras

Orthographic projection



Parallel projection



Weak-perspective projection model

$$\boldsymbol{p} = \frac{1}{z_r}\mathcal{M}\boldsymbol{P}$$

(*p* and *P* are in homogeneous coordinates)

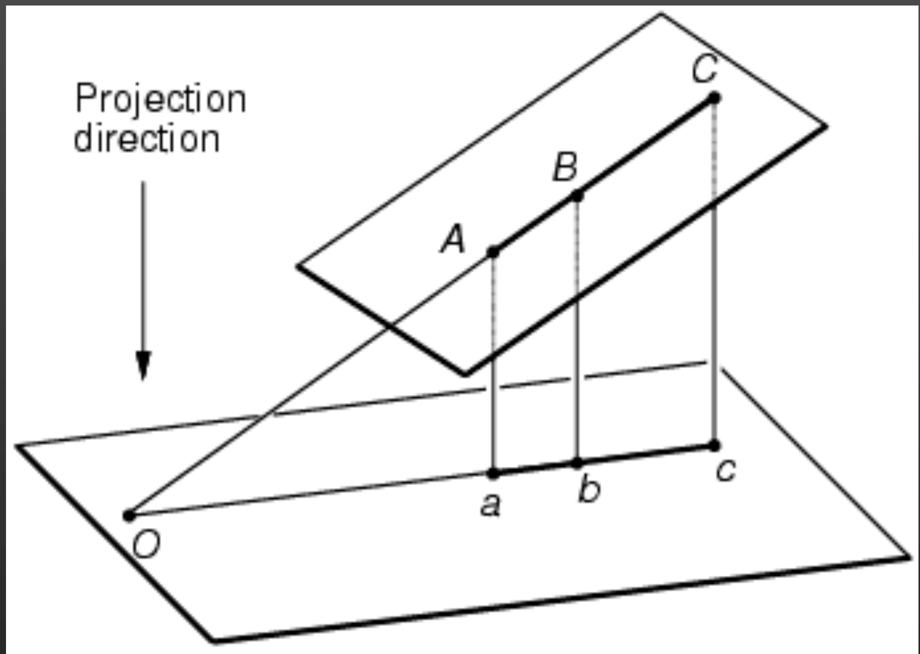
$$p = M\,P$$

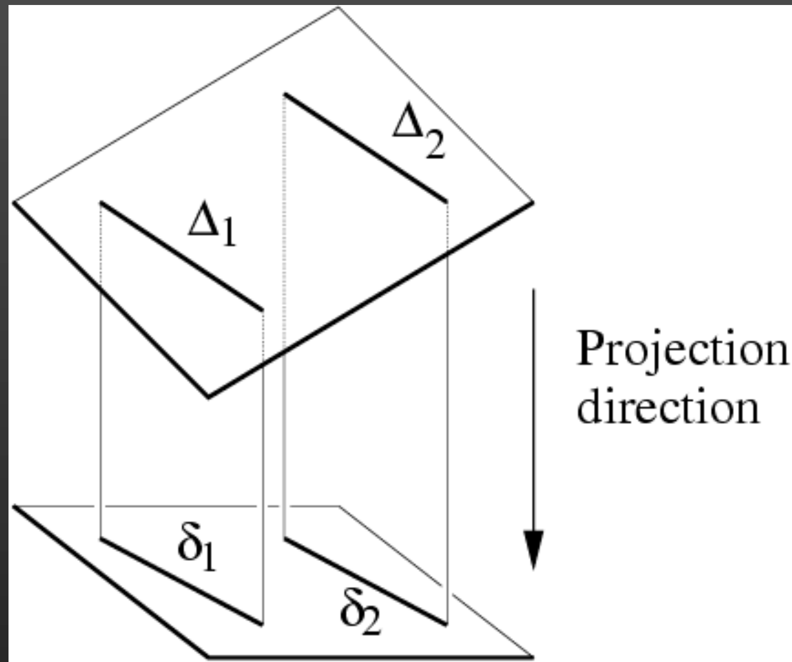
(*P* is in homogeneous coordinates)

$$p = A\,P + b$$

(neither *p* nor *P* is in hom. coordinates)

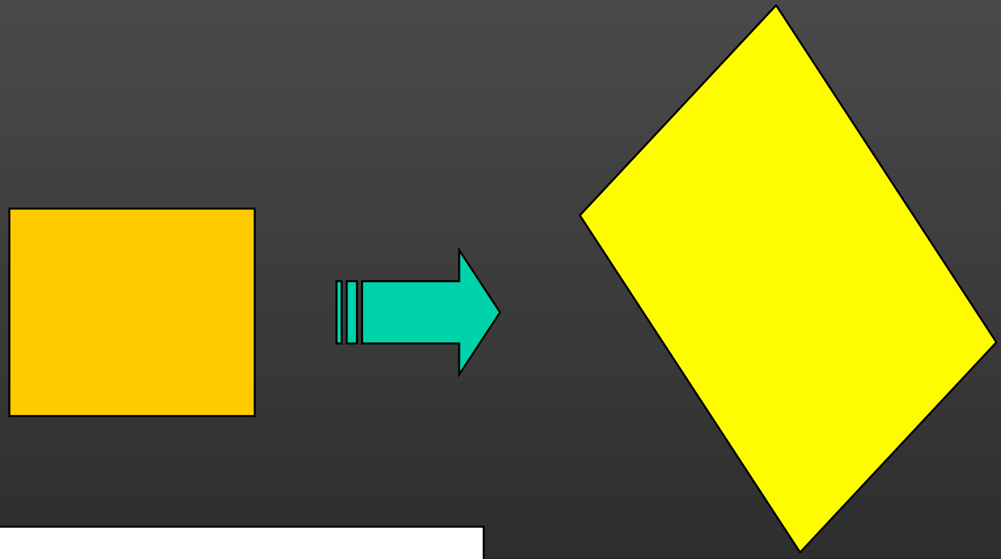
Affine projections induce affine transformations from planes onto their images.





Affine transformations

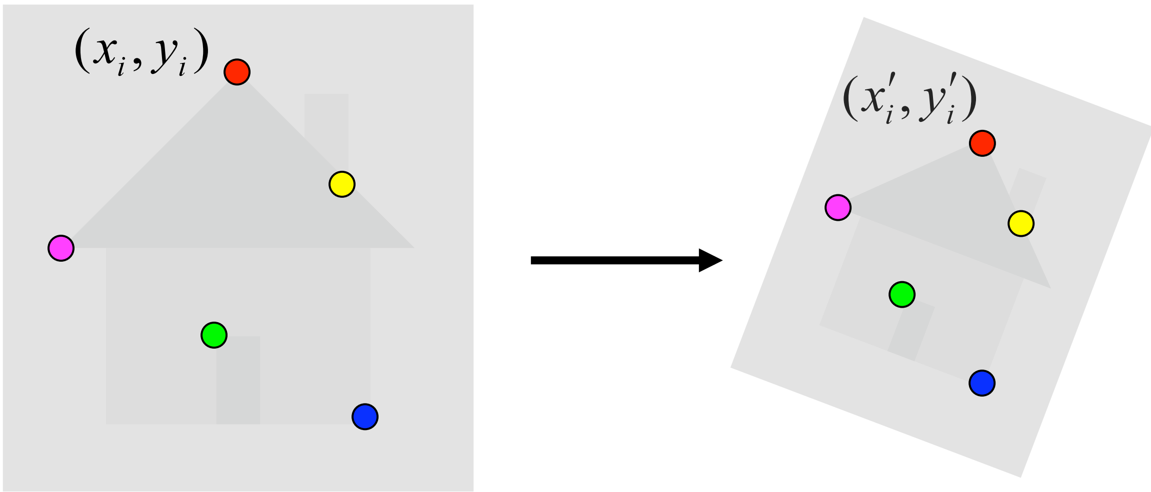
An affine transformation maps a parallelogram onto another parallelogram



$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

Fitting an affine transformation

Assume we know the correspondences, how do we get the transformation?



$$\begin{bmatrix} x_i' \\ y_i' \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

$$\begin{bmatrix} & & \cdots & & & \\ x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \\ & & \cdots & & & \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} \cdots \\ x_i' \\ y_i' \\ \cdots \end{bmatrix}$$

Fitting an affine transformation

$$\begin{bmatrix} & & \cdots & & & \\ x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \\ & & \cdots & & & \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} \cdots \\ x_i' \\ y_i' \\ \cdots \end{bmatrix}$$

Linear system with six unknowns

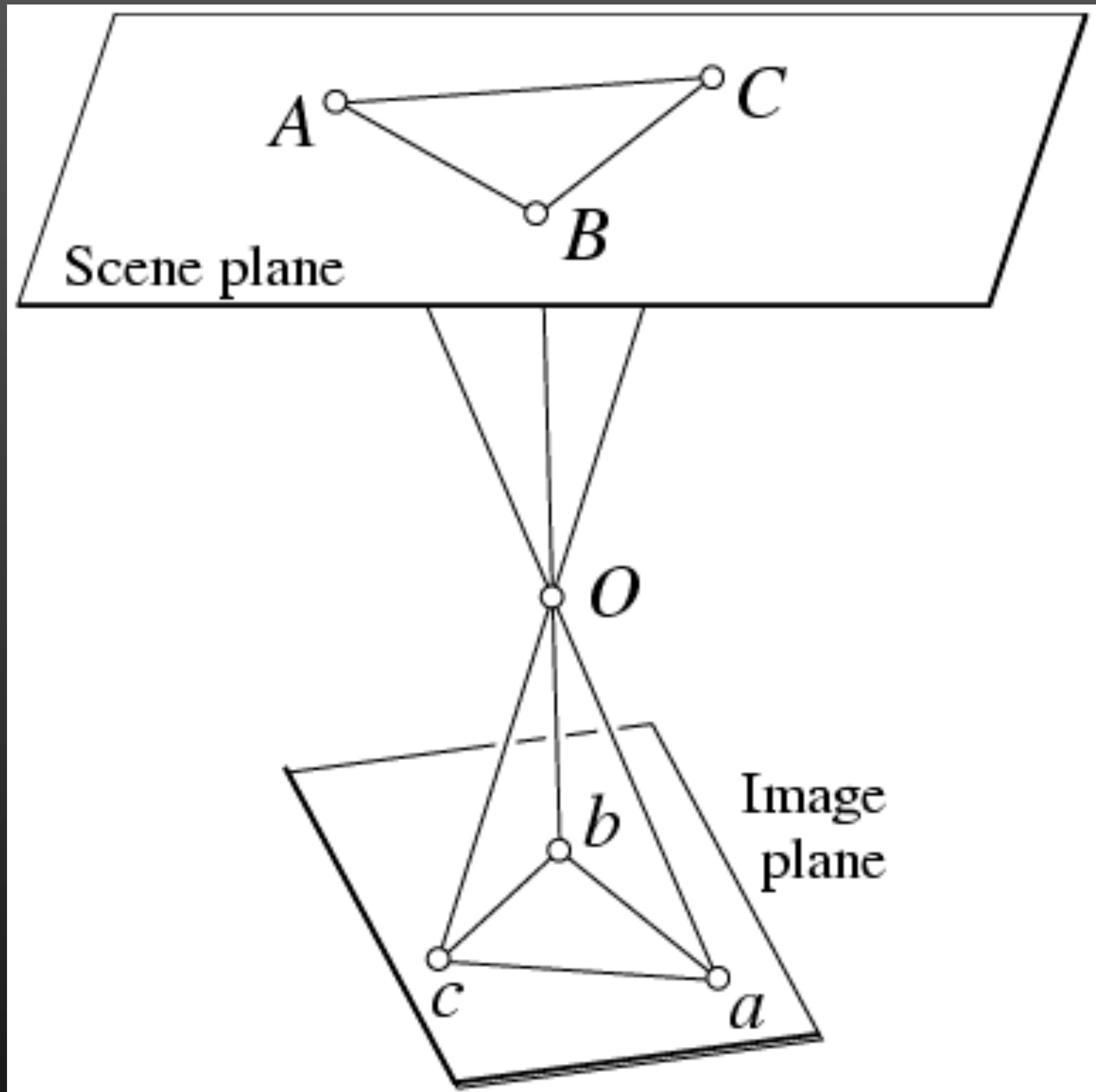
Each match gives us two linearly independent equations: need at least three to solve for the transformation parameters

Beyond affine transformations

What is the transformation between two views of a planar surface?

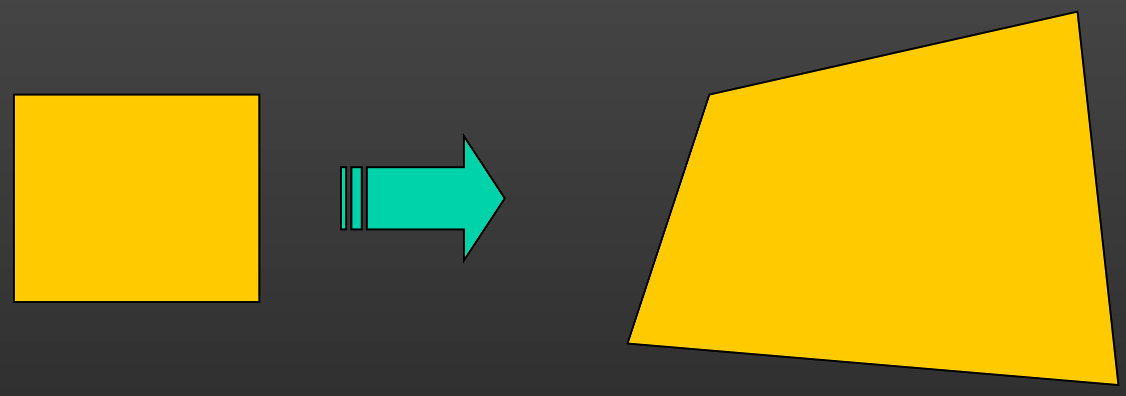
What is the transformation between images from two cameras that share the same center?

Perspective projections induce projective transformations between planes



Beyond affine transformations

Homography: plane projective transformation (transformation taking a quad to another arbitrary quad)



Fitting a homography

Recall: homogenenous coordinates

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Converting *to* homogenenous image coordinates

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w)$$

Converting *from* homogenenous image coordinates

Fitting a homography

Recall: homogenenous coordinates

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Converting *to* homogenenous image coordinates

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w)$$

Converting *from* homogenenous image coordinates

Equation for homography:

$$\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Fitting a homography

Equation for homography:

$$\lambda \begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$$

$$\lambda \mathbf{x}_i' = \mathbf{H}\,\mathbf{x}_i = \begin{bmatrix} \mathbf{h}_1^T \\ \mathbf{h}_2^T \\ \mathbf{h}_3^T \end{bmatrix} \mathbf{x}_i$$

9 entries, 8 degrees of freedom
(scale is arbitrary)

$$\mathbf{x}_i' \times \mathbf{H}\,\mathbf{x}_i = 0$$

$$\mathbf{x}_i' \times \mathbf{H}\,\mathbf{x}_i = \begin{bmatrix} y_i'\,\mathbf{h}_3^T\mathbf{x}_i - \mathbf{h}_2^T\mathbf{x}_i \\ \mathbf{h}_1^T\mathbf{x}_i - x_i'\,\mathbf{h}_3^T\mathbf{x}_i \\ x_i'\,\mathbf{h}_2^T\mathbf{x}_i - y_i'\,\mathbf{h}_1^T\mathbf{x}_i \end{bmatrix}$$

$$\begin{bmatrix} 0^T & -\mathbf{x}_i^T & y_i'\,\mathbf{x}_i^T \\ \mathbf{x}_i^T & 0^T & -x_i'\,\mathbf{x}_i^T \\ -y_i'\,\mathbf{x}_i^T & x_i'\,\mathbf{x}_i^T & 0^T \end{bmatrix} \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \end{pmatrix} = 0$$

3 equations, only 2 linearly independent

Direct linear transform

$$\begin{bmatrix} 0^T & \mathbf{x}_1^T & -y_1' \mathbf{x}_1^T \\ \mathbf{x}_1^T & 0^T & -x_1' \mathbf{x}_1^T \\ \cdots & \cdots & \cdots \\ 0^T & \mathbf{x}_n^T & -y_n' \mathbf{x}_n^T \\ \mathbf{x}_n^T & 0^T & -x_n' \mathbf{x}_n^T \end{bmatrix} \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \end{pmatrix} = 0$$

$$\mathbf{A}\,\mathbf{h} = 0$$

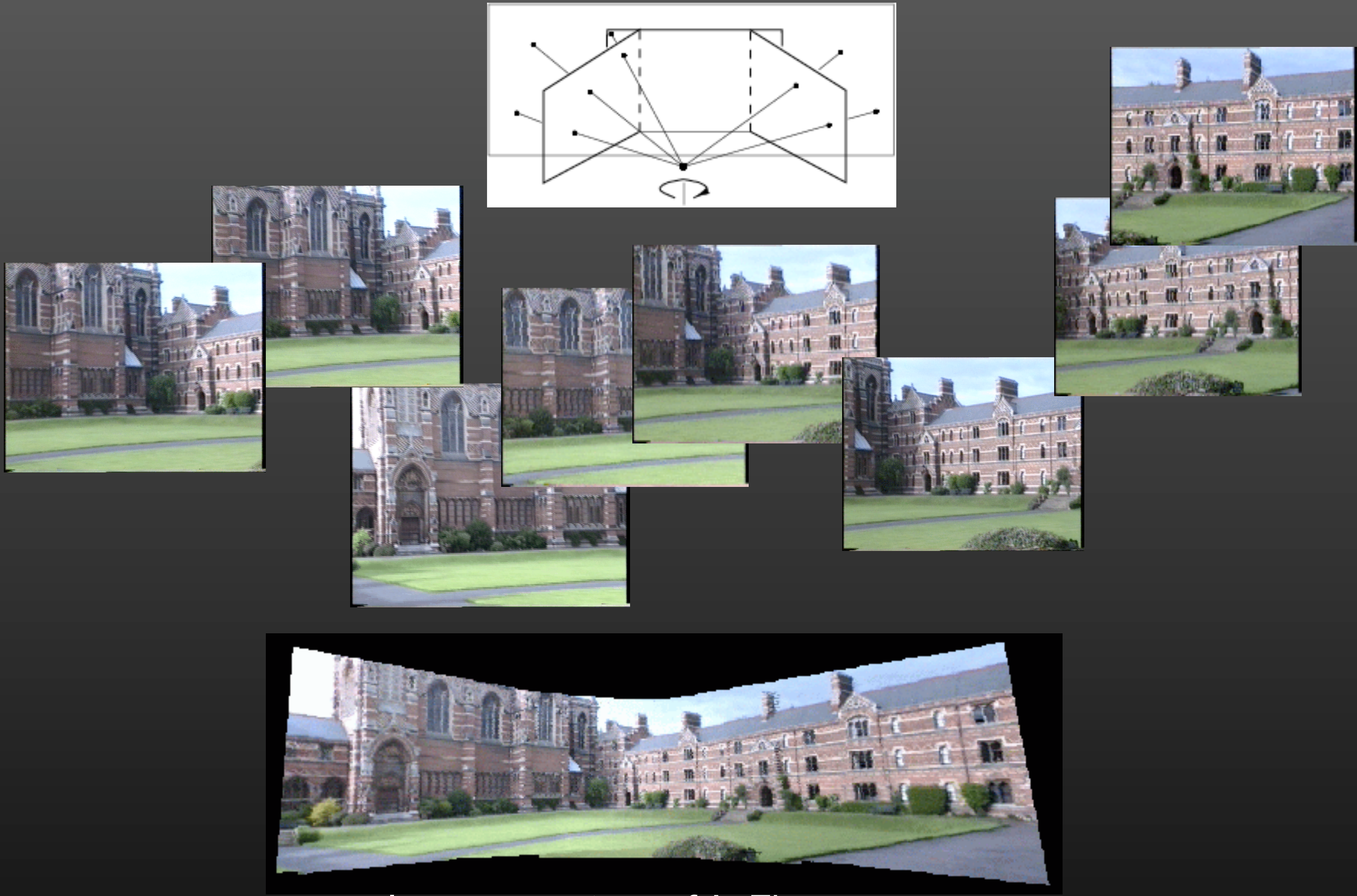
H has 8 degrees of freedom (9 parameters, but scale is arbitrary)

One match gives us two linearly independent equations

Four matches needed for a minimal solution (null space of 8x9 matrix)

More than four: homogeneous least squares

Application: Panorama stitching



Images courtesy of A. Zisserman.