

Computer Vision

CS-E4850, 5 study credits

Lecturer: Juho Kannala

Lecture 8: Camera calibration & single view metrology

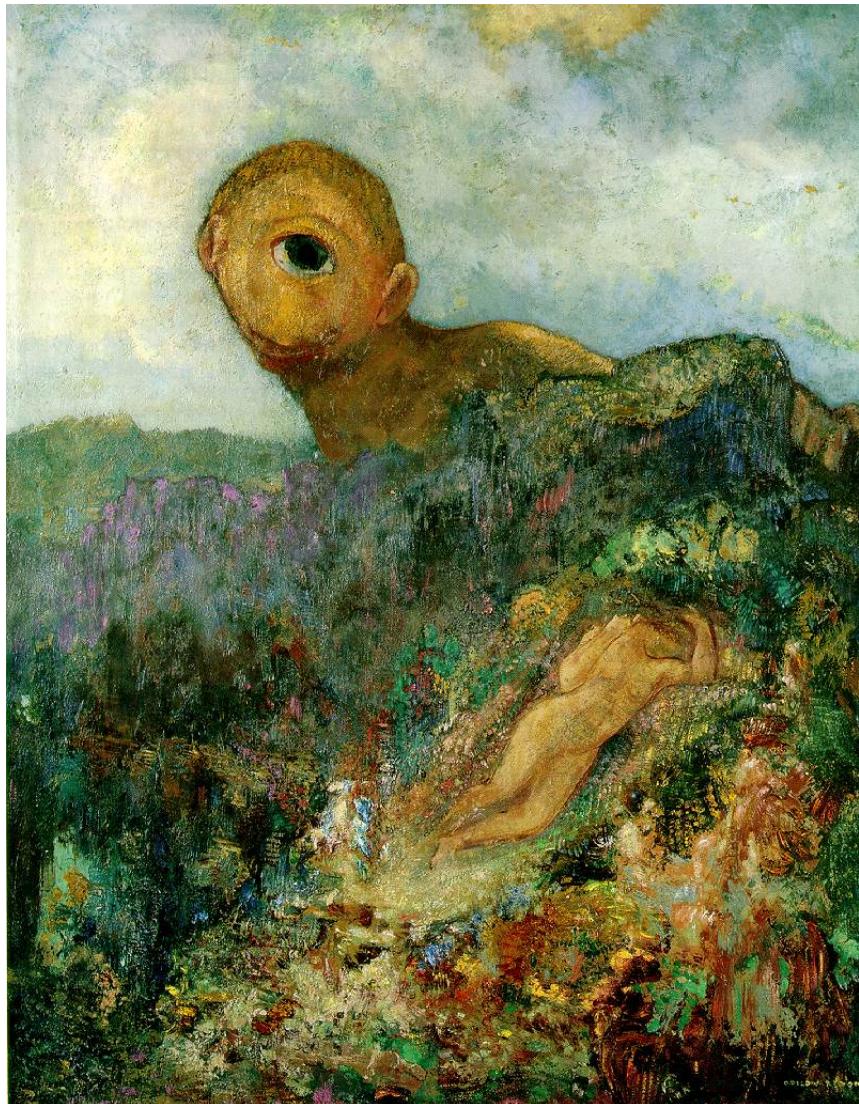
- **Camera calibration** is the process of determining the internal camera parameters, which define the mapping between incoming light rays and image pixels
- **Single view metrology** provides methods for measuring relative lengths from a single image by utilizing certain assumptions

Acknowledgement: many slides from Svetlana Lazebnik, Derek Hoiem, Steve Seitz, and others (detailed credits on individual slides)

Reading

- Szeliski's book, Sections 6.2 and 6.3
- Hartley & Zisserman book, Chapters 6, 7, and 8

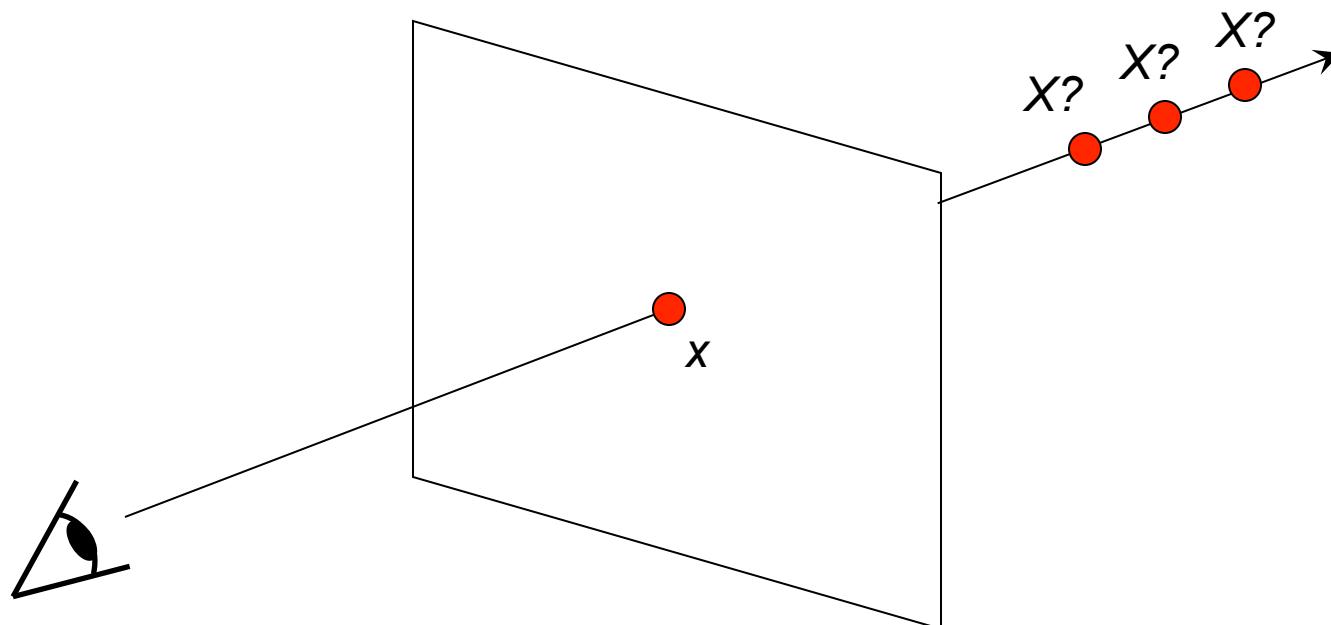
Calibrating a single camera



Odilon Redon, *Cyclops*, 1914

Our goal: Recovery of 3D structure

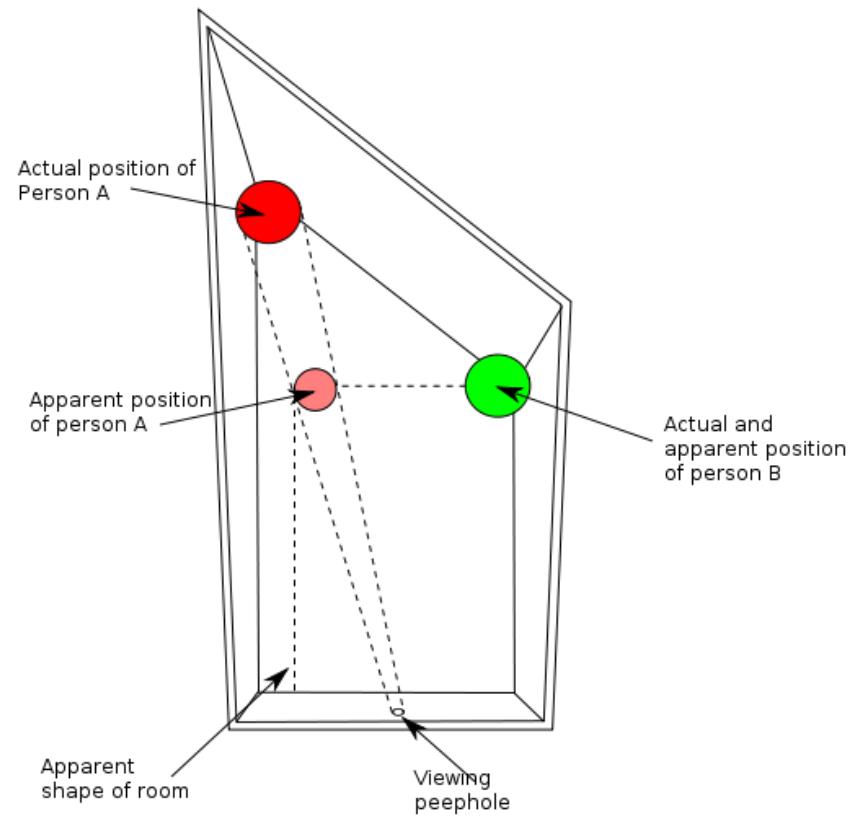
- Recovery of structure from one image is inherently ambiguous



Single-view ambiguity



Single-view ambiguity



Ames room

Our goal: Recovery of 3D structure

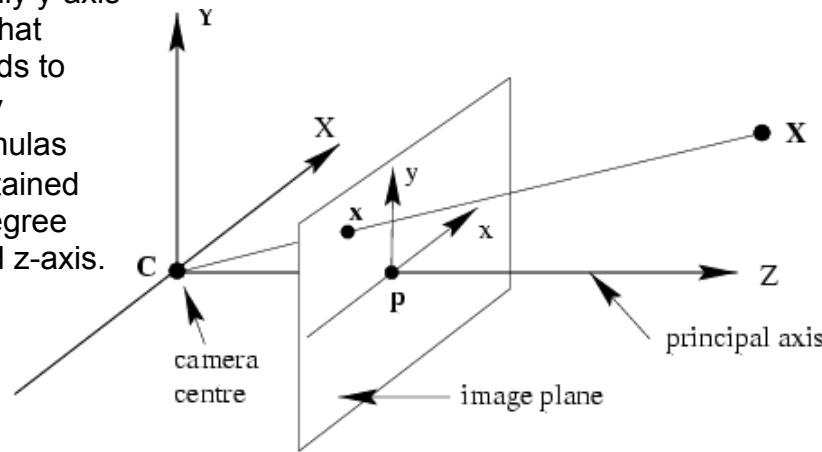
- We will need *multi-view geometry*



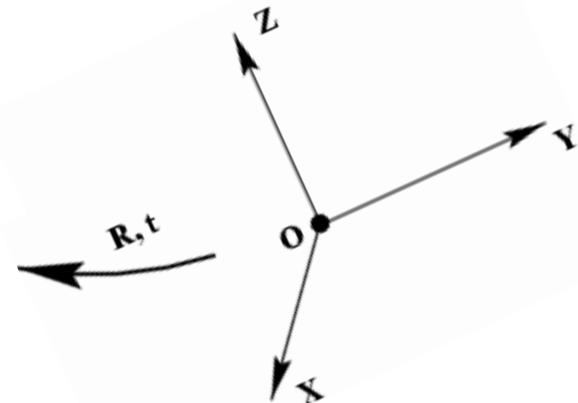
Review: Pinhole camera model

Note that usually y-axis points down. That convention leads to mathematically equivalent formulas and can be obtained here by 180 degree rotation around z-axis.

-->

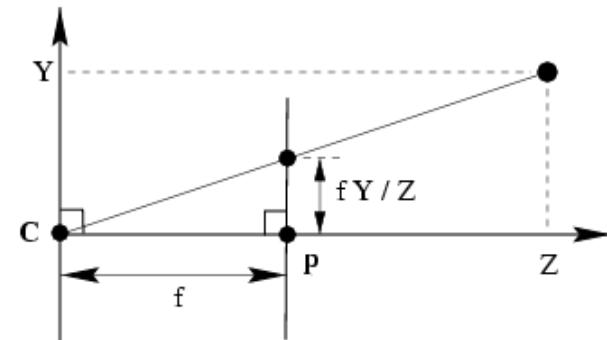
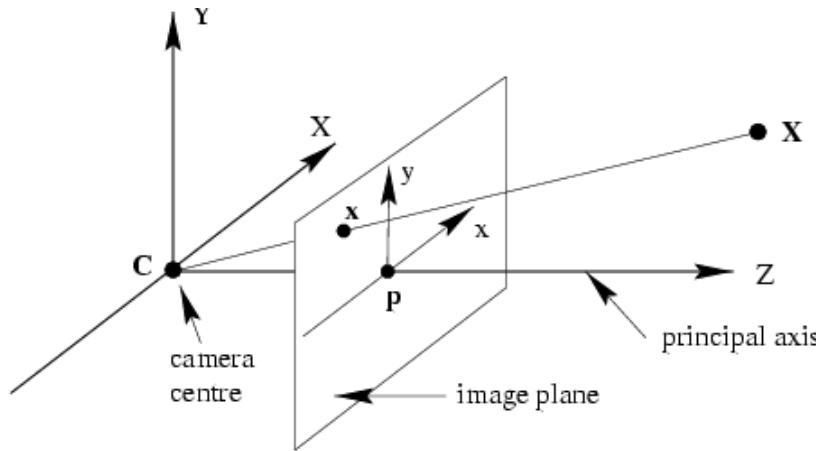


world coordinate system



- **Normalized (camera) coordinate system:** camera center is at the origin, the *principal axis* is the z -axis, x and y axes of the image plane are parallel to x and y axes of the world
- Goal of camera calibration: go from *world* coordinate system to *image* coordinate system

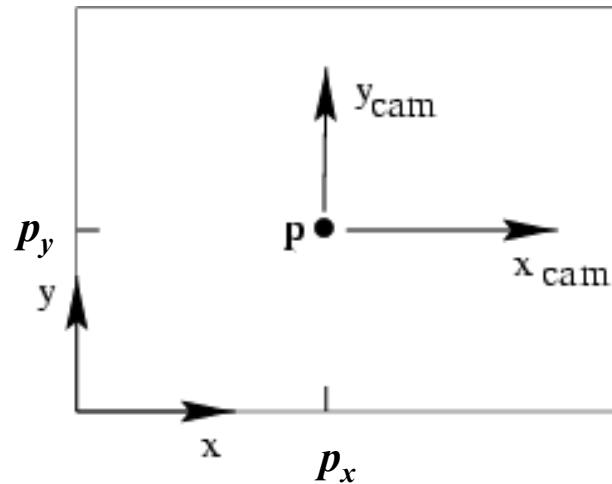
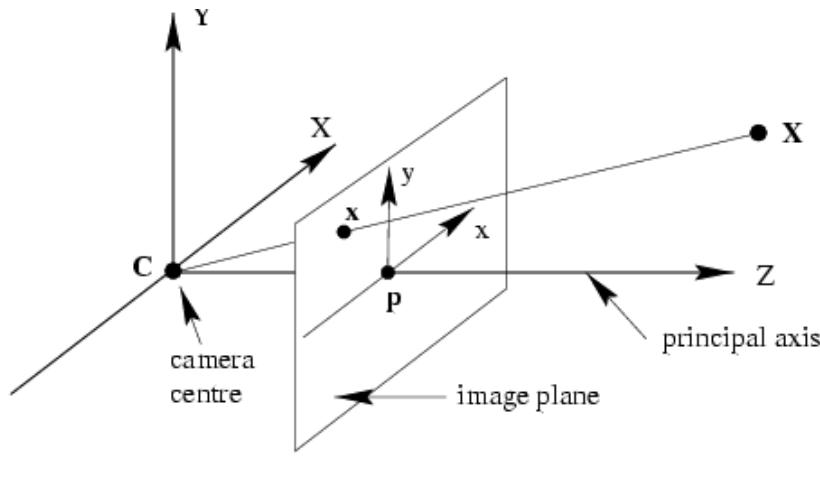
Review: Pinhole camera model



$$(X, Y, Z) \mapsto (fX/Z, fY/Z)$$

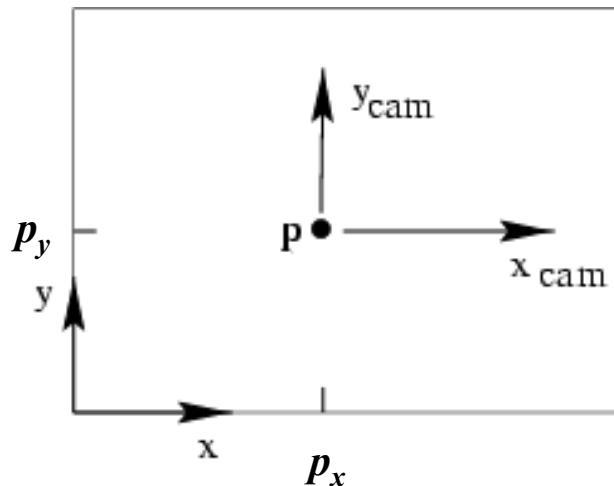
$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad \mathbf{X} = \mathbf{P}\mathbf{X}$$

Principal point



- **Principal point (p):** point where principal axis intersects the image plane
- Normalized coordinate system: origin of the image is at the principal point
- Image coordinate system: origin is in the corner

Principal point offset

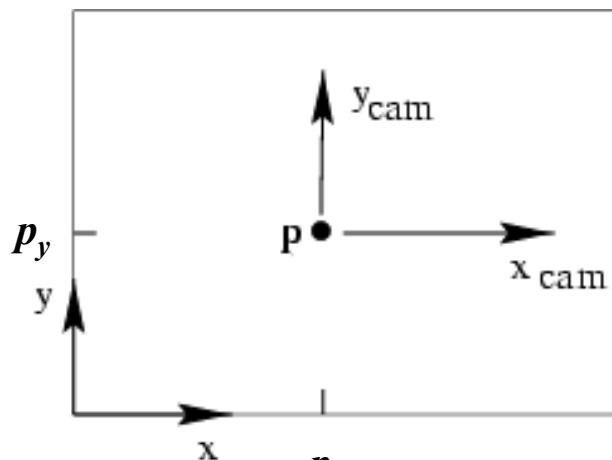


We want the principal point to map to (p_x, p_y) instead of $(0,0)$

$$(X, Y, Z) \mapsto (fX/Z + p_x, fY/Z + p_y)$$

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} f & p_x & 0 \\ f & p_y & 0 \\ 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Principal point offset



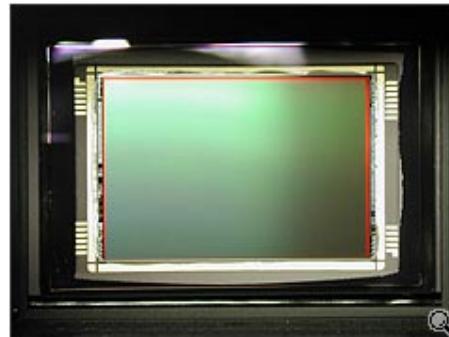
principal point: (p_x, p_y)

$$\begin{pmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{pmatrix} = \begin{bmatrix} f & p_x \\ f & p_y \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$K = \begin{bmatrix} f & p_x \\ f & p_y \\ 1 \end{bmatrix} \quad \text{calibration matrix}$$

$$P = K[I \mid 0]$$

Pixel coordinates



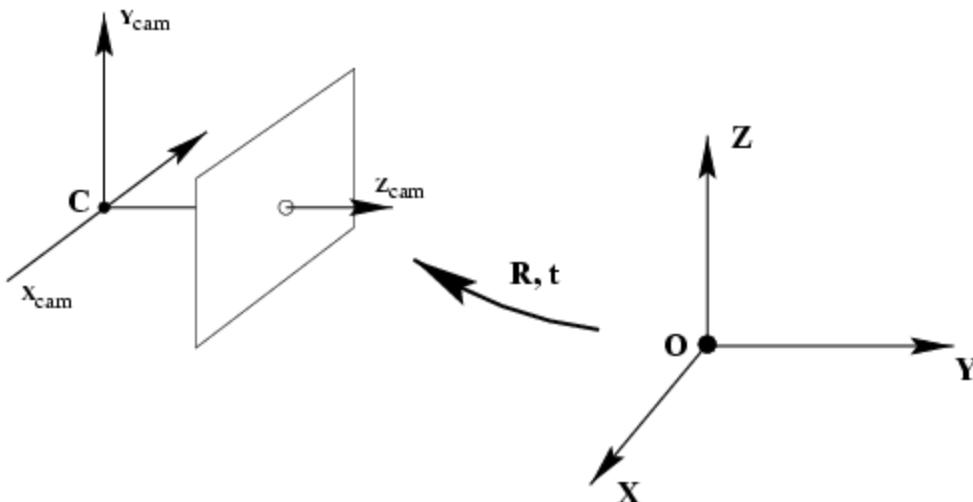
Pixel size: $\frac{1}{m_x} \times \frac{1}{m_y}$

m_x pixels per meter in horizontal direction,
 m_y pixels per meter in vertical direction

$$K = \begin{bmatrix} m_x & & \\ & m_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} f & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & \beta_x \\ \alpha_y & \beta_y \\ & 1 \end{bmatrix}$$

pixels/m m pixels

Camera rotation and translation



- In general, the *camera* coordinate frame will be related to the *world* coordinate frame by a rotation and a translation
- Conversion from world to camera coordinate system (in non-homogeneous coordinates):

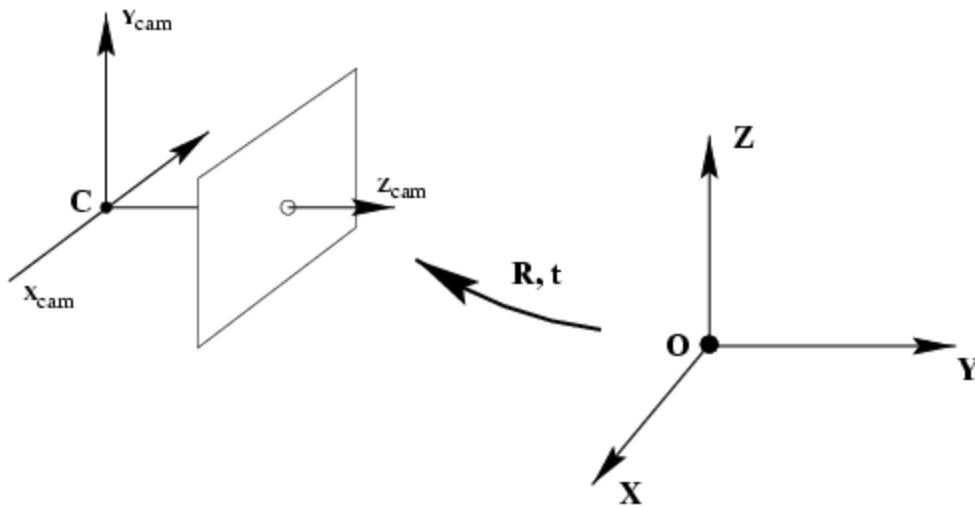
$$\tilde{X}_{\text{cam}} = R(\tilde{X} - \tilde{C})$$

coords. of a point in world frame

coords. of point in camera frame

coords. of camera center in world frame

Camera rotation and translation



$$\tilde{X}_{cam} = R(\tilde{X} - \tilde{C})$$

$$X_{cam} = \begin{pmatrix} \tilde{X}_{cam} \\ 1 \end{pmatrix} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} \tilde{X} \\ 1 \end{pmatrix} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} X$$

$$x = K[I | 0]X_{cam} = K[R | -R\tilde{C}]X \quad P = K[R | t] \quad t = -R\tilde{C}$$

Camera parameters

$$\mathbf{P} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]$$

- Intrinsic parameters
 - Principal point coordinates
 - Focal length
 - Pixel magnification factors
 - *Skew (non-rectangular pixels)*
 - *Radial distortion*

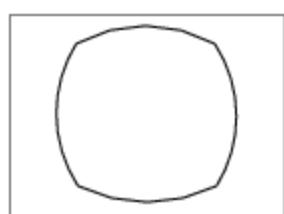
$$\mathbf{K} = \begin{bmatrix} m_x & & \\ & m_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} f & p_x \\ f & p_y \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & \beta_x \\ \alpha_y & \beta_y \\ 1 \end{bmatrix}$$



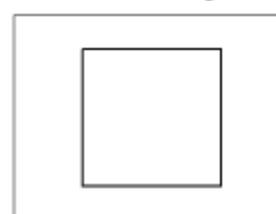
radial distortion



linear image



correction



Camera parameters

$$\mathbf{P} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]$$

- Intrinsic parameters
 - Principal point coordinates
 - Focal length
 - Pixel magnification factors
 - *Skew (non-rectangular pixels)*
 - *Radial distortion*
- Extrinsic parameters
 - Rotation and translation relative to world coordinate system
 - What is the projection of the camera center?

$$\mathbf{P} = \mathbf{K}[\mathbf{R} \quad -\tilde{\mathbf{R}}\tilde{\mathbf{C}}]$$

↑
coords. of
camera center
in world frame

$$\mathbf{P}\mathbf{C} = \mathbf{K}[\mathbf{R} \quad -\tilde{\mathbf{R}}\tilde{\mathbf{C}}]\begin{bmatrix} \tilde{\mathbf{C}} \\ 1 \end{bmatrix} = 0$$

The camera center is the *null space* of the projection matrix!

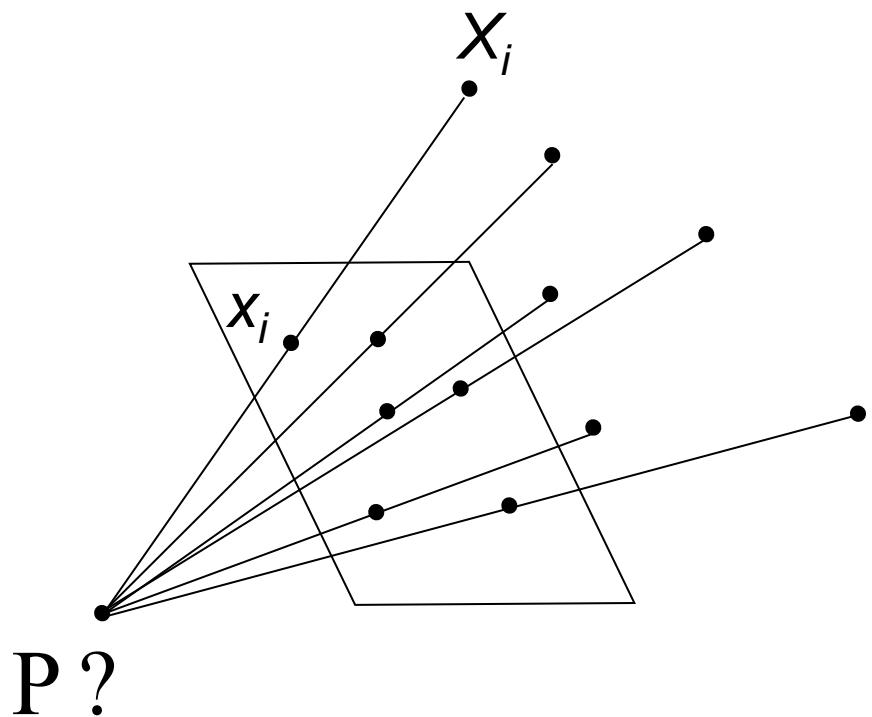
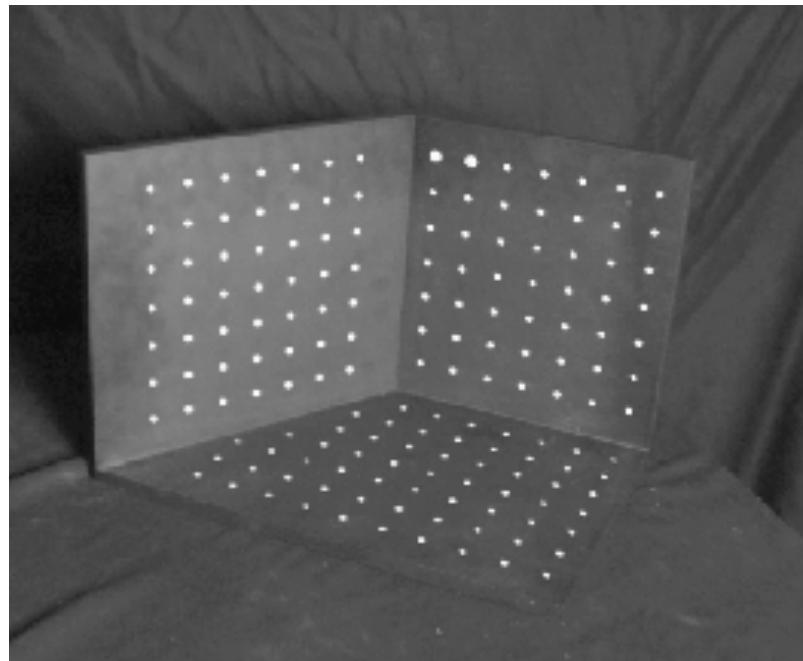
Camera calibration

$$\mathbf{x} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}] \mathbf{X}$$

$$\begin{bmatrix} \lambda_x \\ \lambda_y \\ \lambda \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Camera calibration

- Given n points with known 3D coordinates \mathbf{X}_i and known image projections \mathbf{x}_i , estimate the camera parameters



Camera calibration: Linear method

$$\lambda \mathbf{x}_i = \mathbf{P} \mathbf{X}_i$$

$$\mathbf{x}_i \times \mathbf{P} \mathbf{X}_i = 0$$

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \times \begin{bmatrix} \mathbf{P}_1^T \mathbf{X}_i \\ \mathbf{P}_2^T \mathbf{X}_i \\ \mathbf{P}_3^T \mathbf{X}_i \end{bmatrix} = 0$$

$$\begin{bmatrix} 0 & -\mathbf{X}_i^T & y_i \mathbf{X}_i^T \\ \mathbf{X}_i^T & 0 & -x_i \mathbf{X}_i^T \\ -y_i \mathbf{X}_i^T & x_i \mathbf{X}_i^T & 0 \end{bmatrix} \begin{pmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{pmatrix} = 0$$

Two linearly independent equations

Camera calibration: Linear method

$$\begin{bmatrix} 0^T & \mathbf{X}_1^T & -y_1\mathbf{X}_1^T \\ \mathbf{X}_1^T & 0^T & -x_1\mathbf{X}_1^T \\ \dots & \dots & \dots \\ 0^T & \mathbf{X}_n^T & -y_n\mathbf{X}_n^T \\ \mathbf{X}_n^T & 0^T & -x_n\mathbf{X}_n^T \end{bmatrix} \begin{pmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{pmatrix} = 0 \quad \mathbf{A}\mathbf{p} = \mathbf{0}$$

- \mathbf{P} has 11 degrees of freedom
- One 2D/3D correspondence gives us two linearly independent equations
 - 6 correspondences needed for a minimal solution
- Homogeneous least squares: find \mathbf{p} ($\|\mathbf{p}\|=1$) minimizing $\|\mathbf{Ap}\|^2$
 - Solution given by eigenvector of $\mathbf{A}^T\mathbf{A}$ with smallest eigenvalue

Camera calibration: Linear method

$$\begin{bmatrix} 0^T & \mathbf{X}_1^T & -y_1\mathbf{X}_1^T \\ \mathbf{X}_1^T & 0^T & -x_1\mathbf{X}_1^T \\ \dots & \dots & \dots \\ 0^T & \mathbf{X}_n^T & -y_n\mathbf{X}_n^T \\ \mathbf{X}_n^T & 0^T & -x_n\mathbf{X}_n^T \end{bmatrix} \begin{pmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{pmatrix} = 0 \quad \mathbf{A}\mathbf{p} = \mathbf{0}$$

- Note: for coplanar points that satisfy $\Pi^T \mathbf{X} = 0$, we will get degenerate solutions $(\Pi, 0, 0)$, $(0, \Pi, 0)$, or $(0, 0, \Pi)$

Camera calibration: Linear method

- The linear method only estimates the entries of the projection matrix:

$$\begin{bmatrix} \lambda_x \\ \lambda_y \\ \lambda \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- What we ultimately want is a decomposition of this matrix into the intrinsic and extrinsic parameters:

$$\mathbf{x} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}] \mathbf{X}$$

- This can be achieved via the RQ matrix decomposition (see Sec. 6.2.4 of H&Z book)

Camera calibration: Linear method

- Advantages: easy to formulate and solve
- Disadvantages
 - Doesn't directly tell you camera parameters
 - Doesn't model radial distortion
 - Can't impose constraints, such as known focal length and orthogonality
- Non-linear methods are preferred
 - Define error as sum of squared distances between measured 2D points and estimated projections of 3D points
 - Minimize error using Newton's method or other non-linear optimization
 - The iterative optimization by non-linear methods can be initialized with the solution provided by the linear method

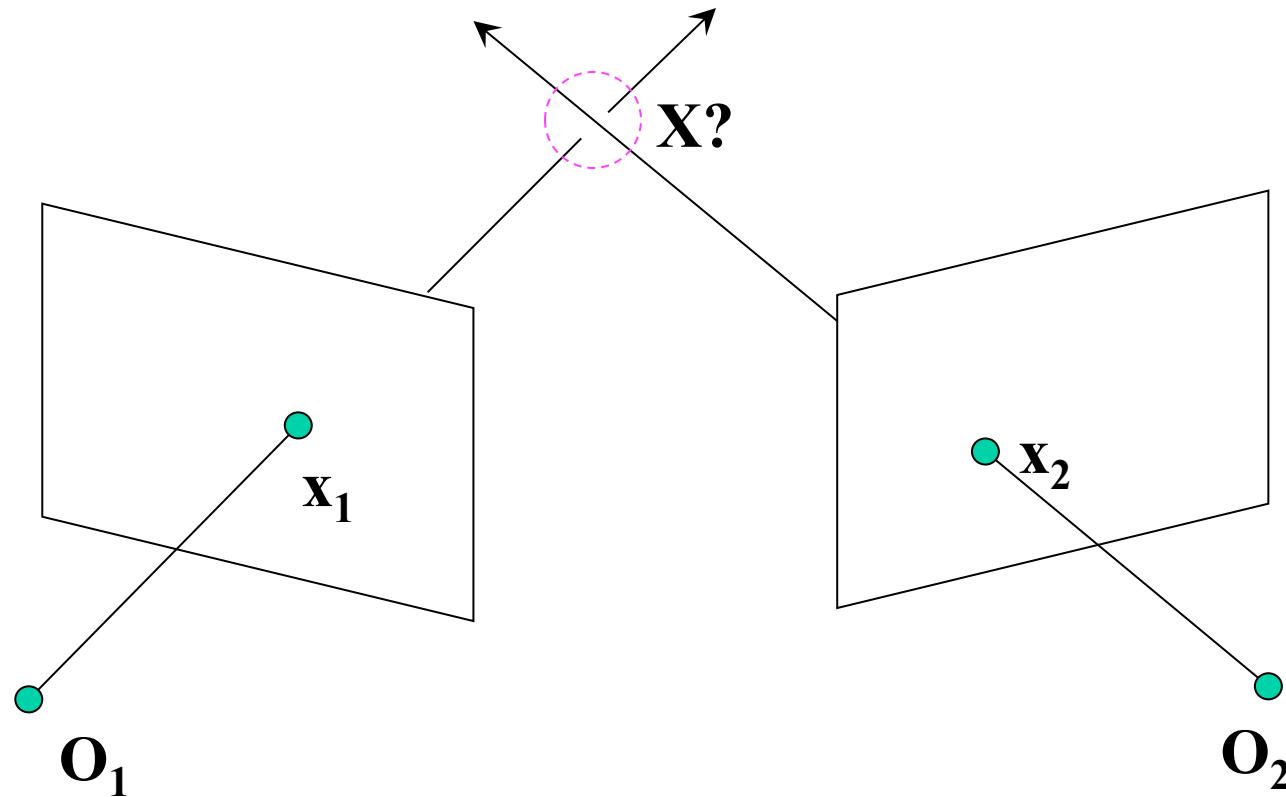
A taste of multi-view geometry: Triangulation

- Given projections of a 3D point in two or more images (with known camera matrices), find the coordinates of the point



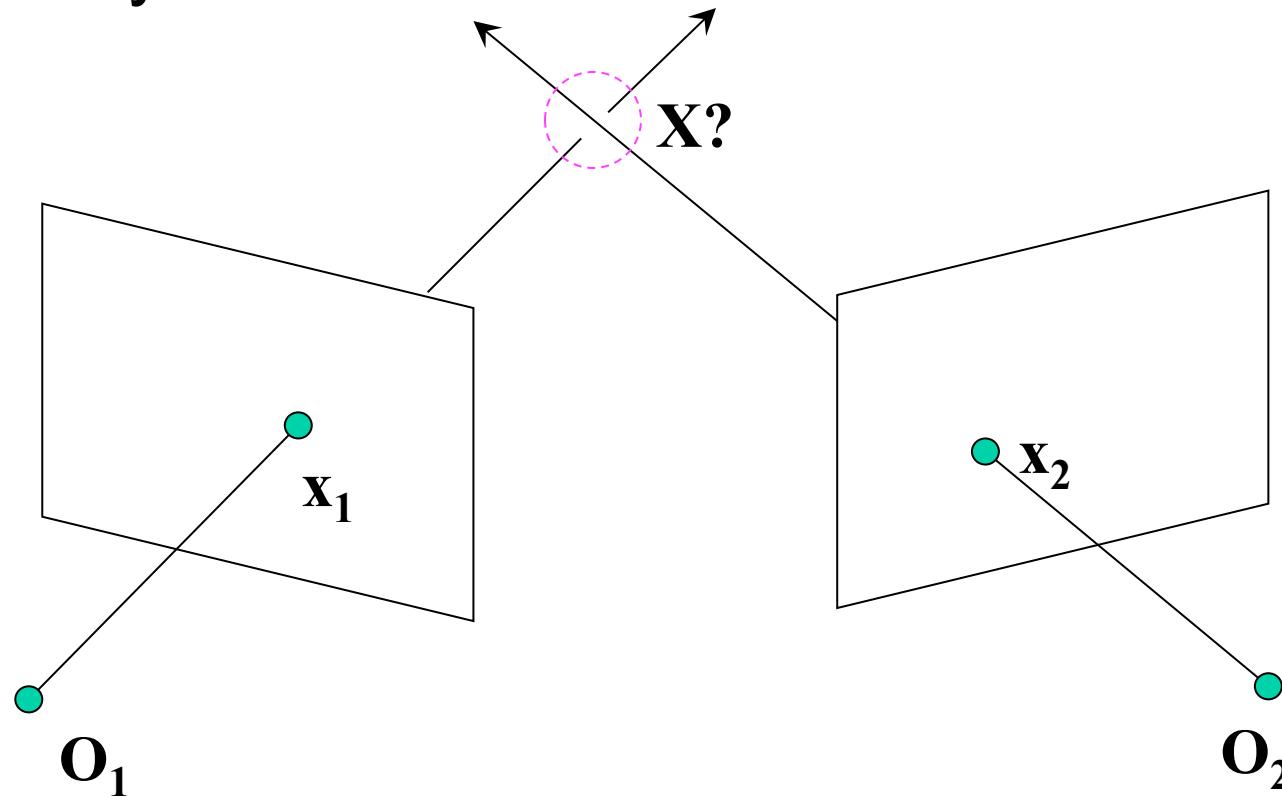
Triangulation

- Given projections of a 3D point in two or more images (with known camera matrices), find the coordinates of the point



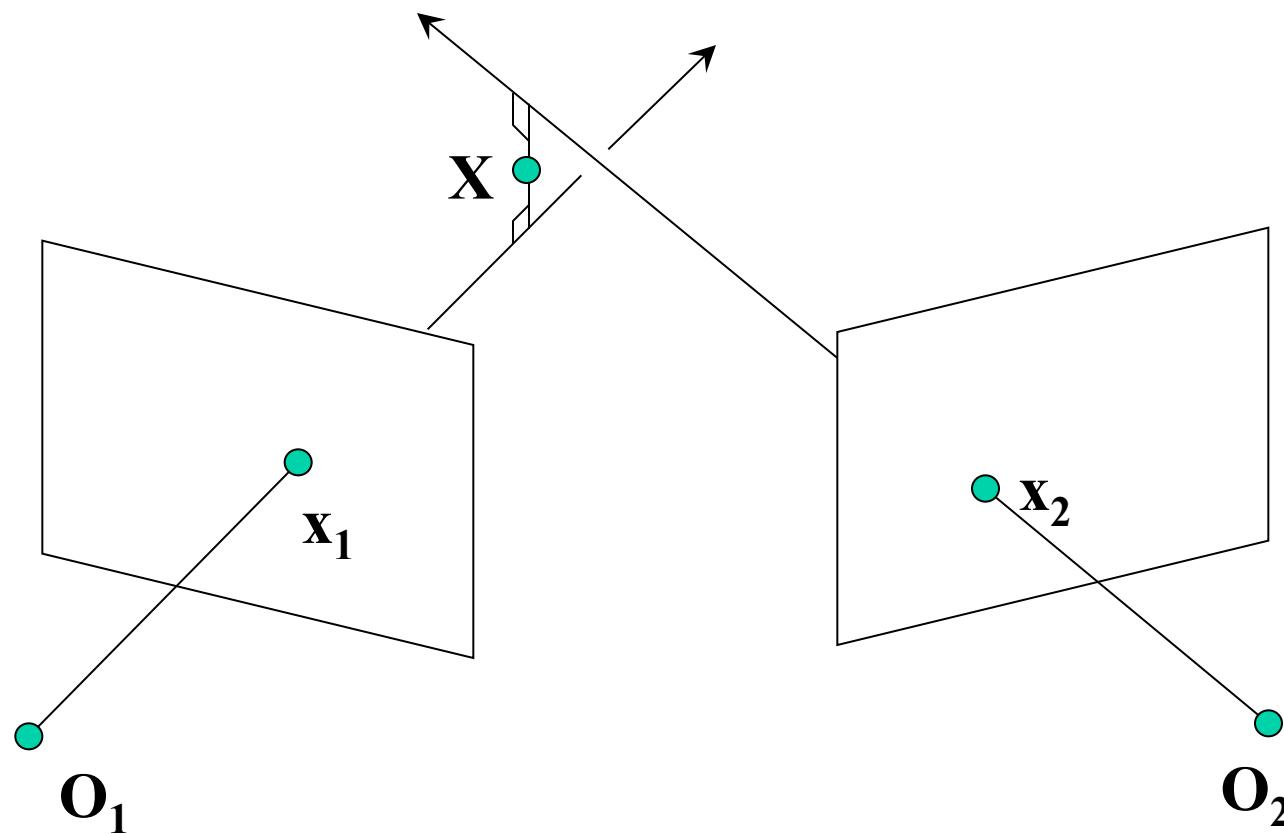
Triangulation

- We want to intersect the two visual rays corresponding to x_1 and x_2 , but because of noise and numerical errors, they don't meet exactly



Triangulation: Geometric approach

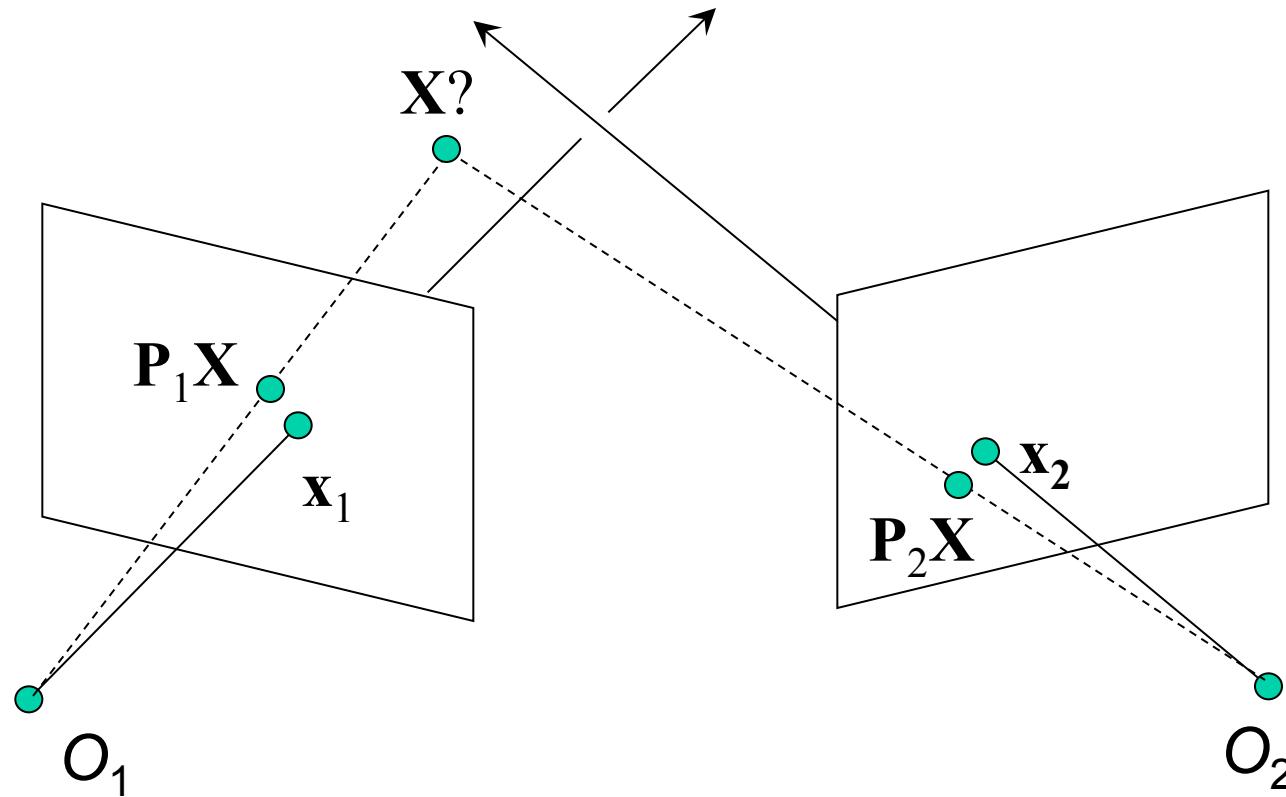
- Find shortest segment connecting the two viewing rays and let X be the midpoint of that segment



Triangulation: Nonlinear approach

Find X that minimizes

$$d^2(\mathbf{x}_1, \mathbf{P}_1\mathbf{X}) + d^2(\mathbf{x}_2, \mathbf{P}_2\mathbf{X})$$



Triangulation: Linear approach

$$\lambda_1 \mathbf{x}_1 = \mathbf{P}_1 \mathbf{X} \quad \mathbf{x}_1 \times \mathbf{P}_1 \mathbf{X} = \mathbf{0} \quad [\mathbf{x}_{1\times}] \mathbf{P}_1 \mathbf{X} = \mathbf{0}$$

$$\lambda_2 \mathbf{x}_2 = \mathbf{P}_2 \mathbf{X} \quad \mathbf{x}_2 \times \mathbf{P}_2 \mathbf{X} = \mathbf{0} \quad [\mathbf{x}_{2\times}] \mathbf{P}_2 \mathbf{X} = \mathbf{0}$$

Cross product as matrix multiplication:

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

Triangulation: Linear approach

$$\lambda_1 \mathbf{x}_1 = \mathbf{P}_1 \mathbf{X} \quad \mathbf{x}_1 \times \mathbf{P}_1 \mathbf{X} = \mathbf{0} \quad [\mathbf{x}_{1\times}] \mathbf{P}_1 \mathbf{X} = \mathbf{0}$$

$$\lambda_2 \mathbf{x}_2 = \mathbf{P}_2 \mathbf{X} \quad \mathbf{x}_2 \times \mathbf{P}_2 \mathbf{X} = \mathbf{0} \quad [\mathbf{x}_{2\times}] \mathbf{P}_2 \mathbf{X} = \mathbf{0}$$



Two independent equations each in terms of
three unknown entries of \mathbf{X}

Single-view metrology

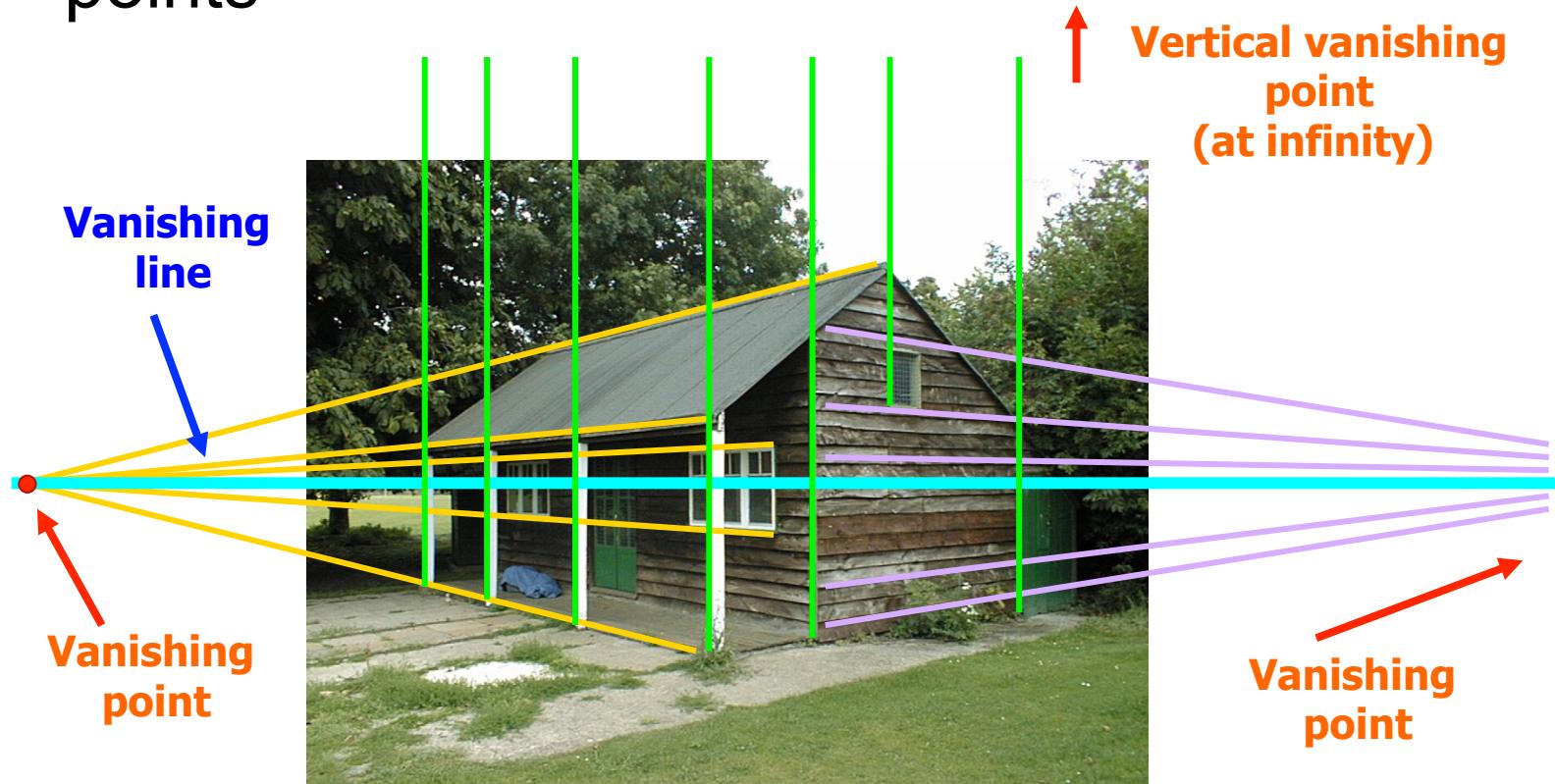


Magritte, *Personal Values*, 1952

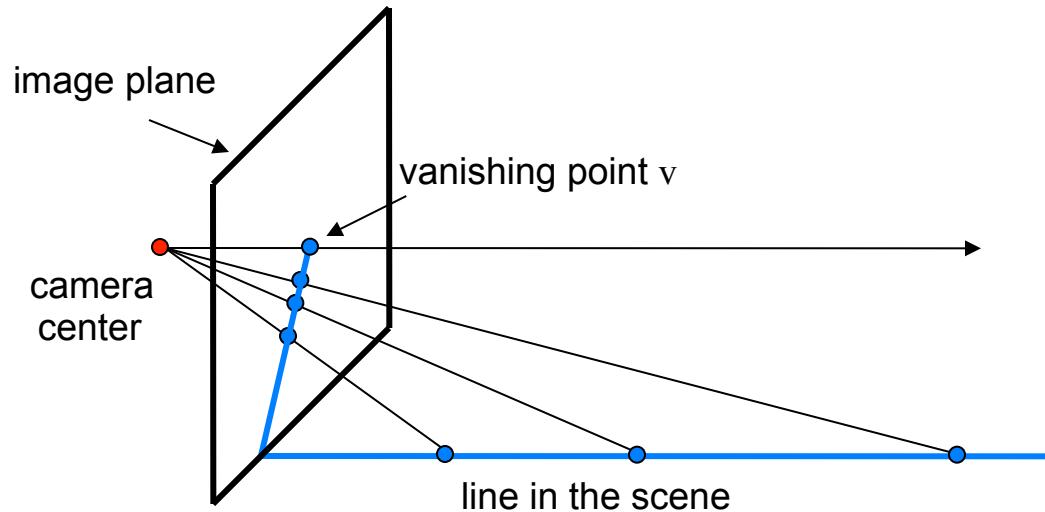
Many slides from S. Seitz, D. Hoiem

Camera calibration revisited

- What if world coordinates of reference 3D points are not known?
- We can use scene features such as vanishing points

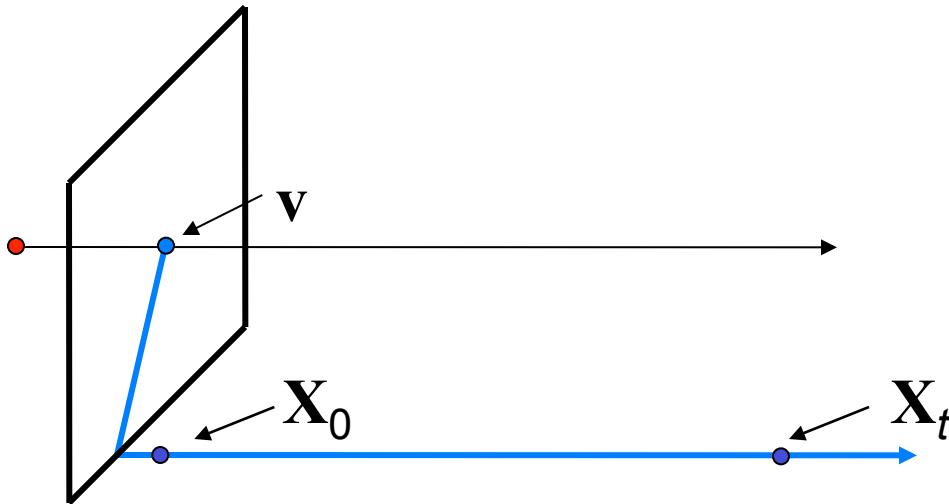


Recall: Vanishing points



- All lines having the same direction share the same vanishing point

Computing vanishing points

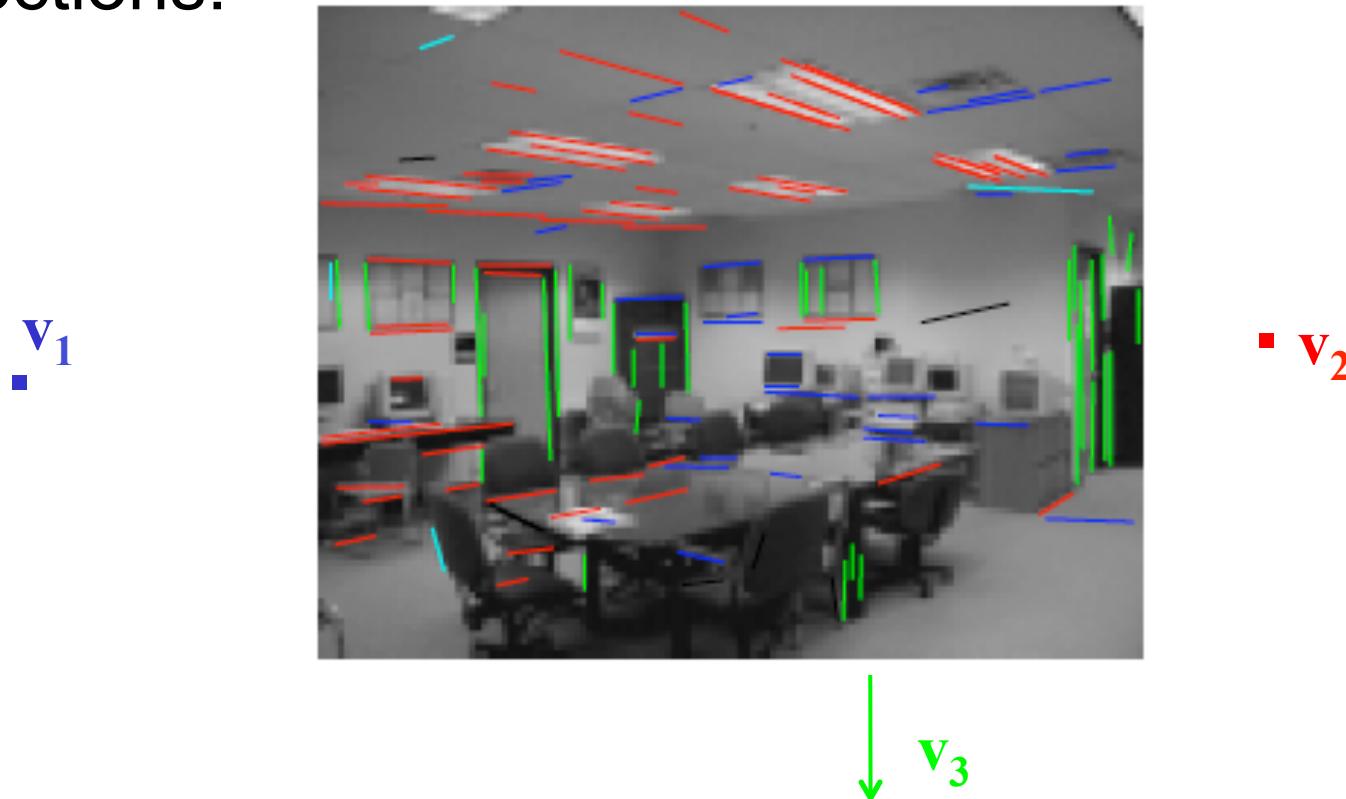


$$\mathbf{X}_t = \begin{bmatrix} x_0 + td_1 \\ y_0 + td_2 \\ z_0 + td_3 \\ 1 \end{bmatrix} = \begin{bmatrix} x_0 / t + d_1 \\ y_0 / t + d_2 \\ z_0 / t + d_3 \\ 1/t \end{bmatrix} \quad \mathbf{X}_{\infty} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ 0 \end{bmatrix}$$

- \mathbf{X}_{∞} is a *point at infinity*, \mathbf{v} is its projection: $\mathbf{v} = \mathbf{P}\mathbf{X}_{\infty}$
- The vanishing point depends only on *line direction*
- All lines having direction \mathbf{d} intersect at \mathbf{X}_{∞}

Calibration from vanishing points

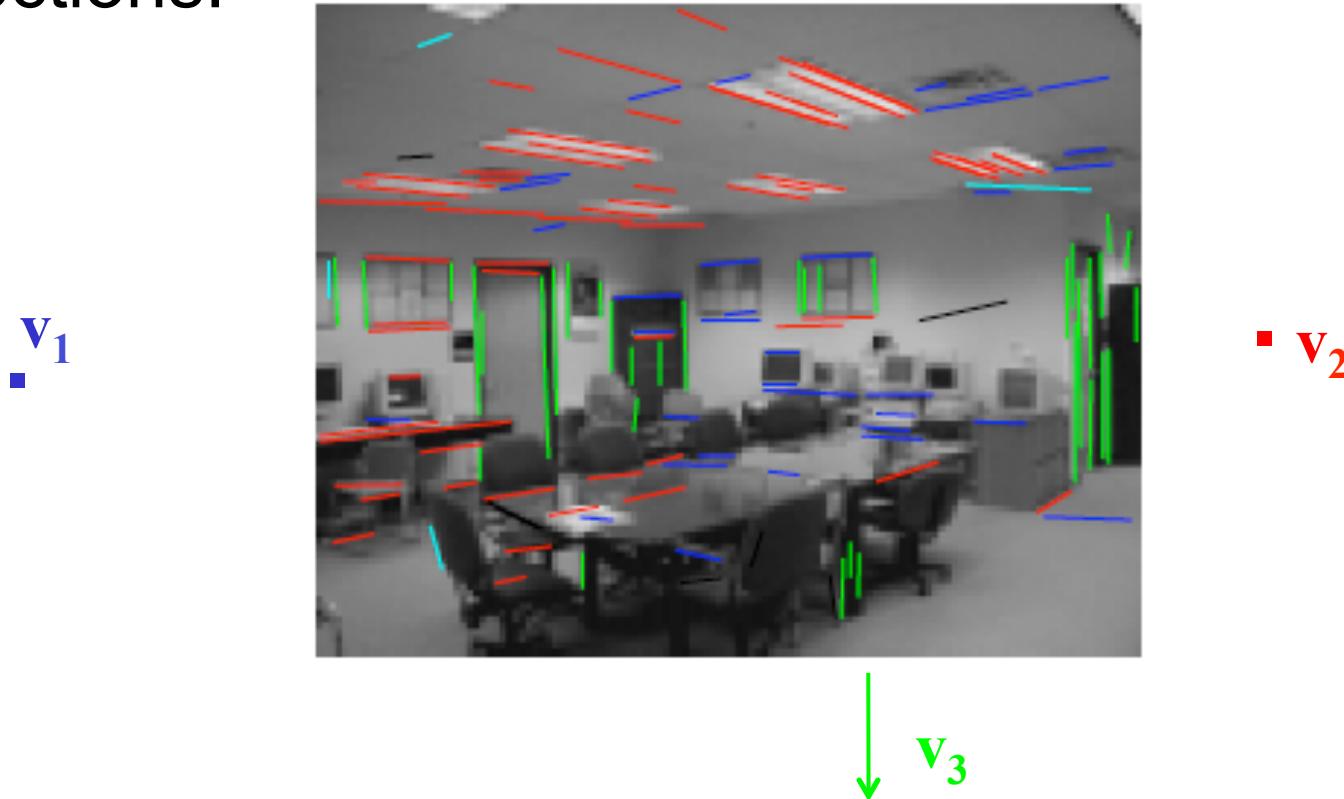
- Consider a scene with three orthogonal vanishing directions:



- Note: v_1 , v_2 are *finite* vanishing points and v_3 is an *infinite* vanishing point

Calibration from vanishing points

- Consider a scene with three orthogonal vanishing directions:



- We can align the world coordinate system with these directions

Calibration from vanishing points

$$\mathbf{P} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} = [\mathbf{p}_1 \quad \mathbf{p}_2 \quad \mathbf{p}_3 \quad \mathbf{p}_4]$$

- $\mathbf{p}_1 = \mathbf{P}(1,0,0,0)^T$ – the vanishing point in the x direction
- Similarly, \mathbf{p}_2 and \mathbf{p}_3 are the vanishing points in the y and z directions
- $\mathbf{p}_4 = \mathbf{P}(0,0,0,1)^T$ – projection of the origin of the world coordinate system
- Problem: we can only know the four columns up to independent scale factors, additional constraints needed to solve for them

Calibration from vanishing points

- Let us align the world coordinate system with three orthogonal vanishing directions in the scene:

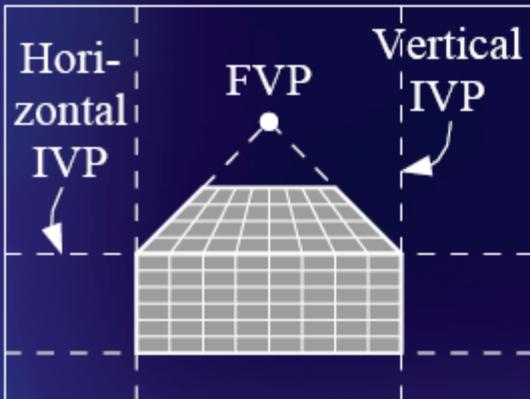
$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \lambda_i \mathbf{v}_i = \mathbf{K} [\mathbf{R} \mid \mathbf{t}] \begin{bmatrix} \mathbf{e}_i \\ 0 \end{bmatrix} = \mathbf{K} \mathbf{R} \mathbf{e}_i$$

$$\mathbf{e}_i = \lambda_i \mathbf{R}^T \mathbf{K}^{-1} \mathbf{v}_i, \quad \mathbf{e}_i^T \mathbf{e}_j = 0$$

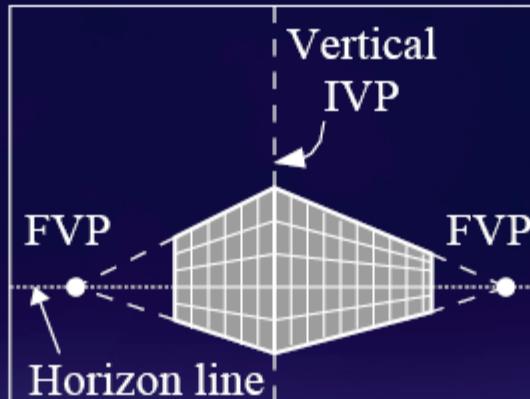
$$\mathbf{v}_i^T \mathbf{K}^{-T} \mathbf{R} \mathbf{R}^T \mathbf{K}^{-1} \mathbf{v}_j = \mathbf{v}_i^T \mathbf{K}^{-T} \mathbf{K}^{-1} \mathbf{v}_j = 0$$

- Each pair of vanishing points gives us a constraint on the focal length and principal point

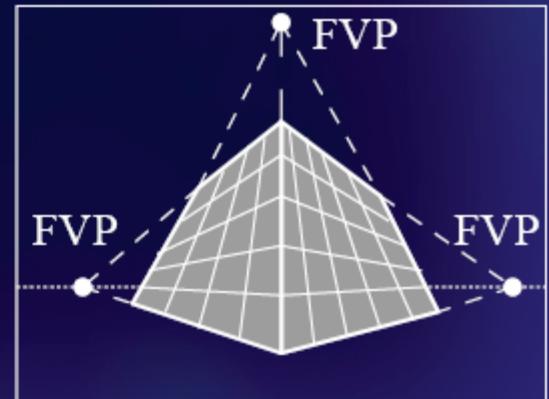
Calibration from vanishing points



1 finite vanishing point,
2 infinite vanishing points



2 finite vanishing points,
1 infinite vanishing point



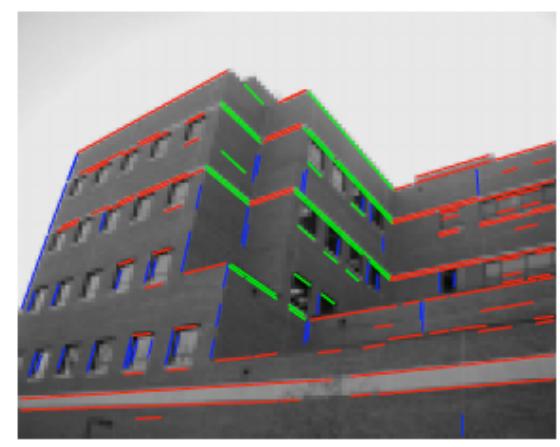
3 finite vanishing points



Cannot recover focal
length, principal point is
the third vanishing point



Can solve for focal length, principal point



Rotation from vanishing points

$$\lambda_i \mathbf{v}_i = \mathbf{K}[\mathbf{R} | \mathbf{t}] \begin{bmatrix} \mathbf{e}_i \\ 0 \end{bmatrix} = \mathbf{K}\mathbf{R}\mathbf{e}_i$$

$$\lambda_1 \mathbf{K}^{-1} \mathbf{v}_1 = \mathbf{R}\mathbf{e}_1 = [\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{r}_3] \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \mathbf{r}_1$$

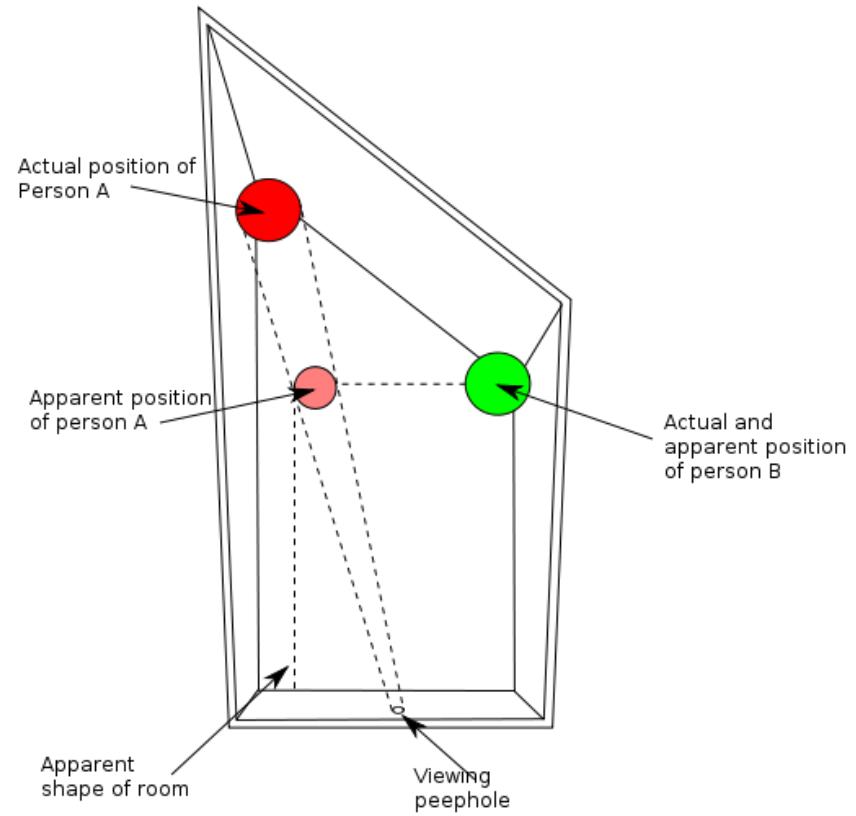
Thus, $\lambda_i \mathbf{K}^{-1} \mathbf{v}_i = \mathbf{r}_i$.

Get λ_i by using the constraint $\|\mathbf{r}_i\|^2=1$.

Calibration from vanishing points: Summary

- Solve for K (focal length, principal point) using three orthogonal vanishing points
- Get rotation directly from vanishing points once calibration matrix is known
- Advantages
 - No need for calibration chart, 2D-3D correspondences
 - Could be completely automatic
- Disadvantages
 - Only applies to certain kinds of scenes
 - Inaccuracies in computation of vanishing points
 - Problems due to infinite vanishing points

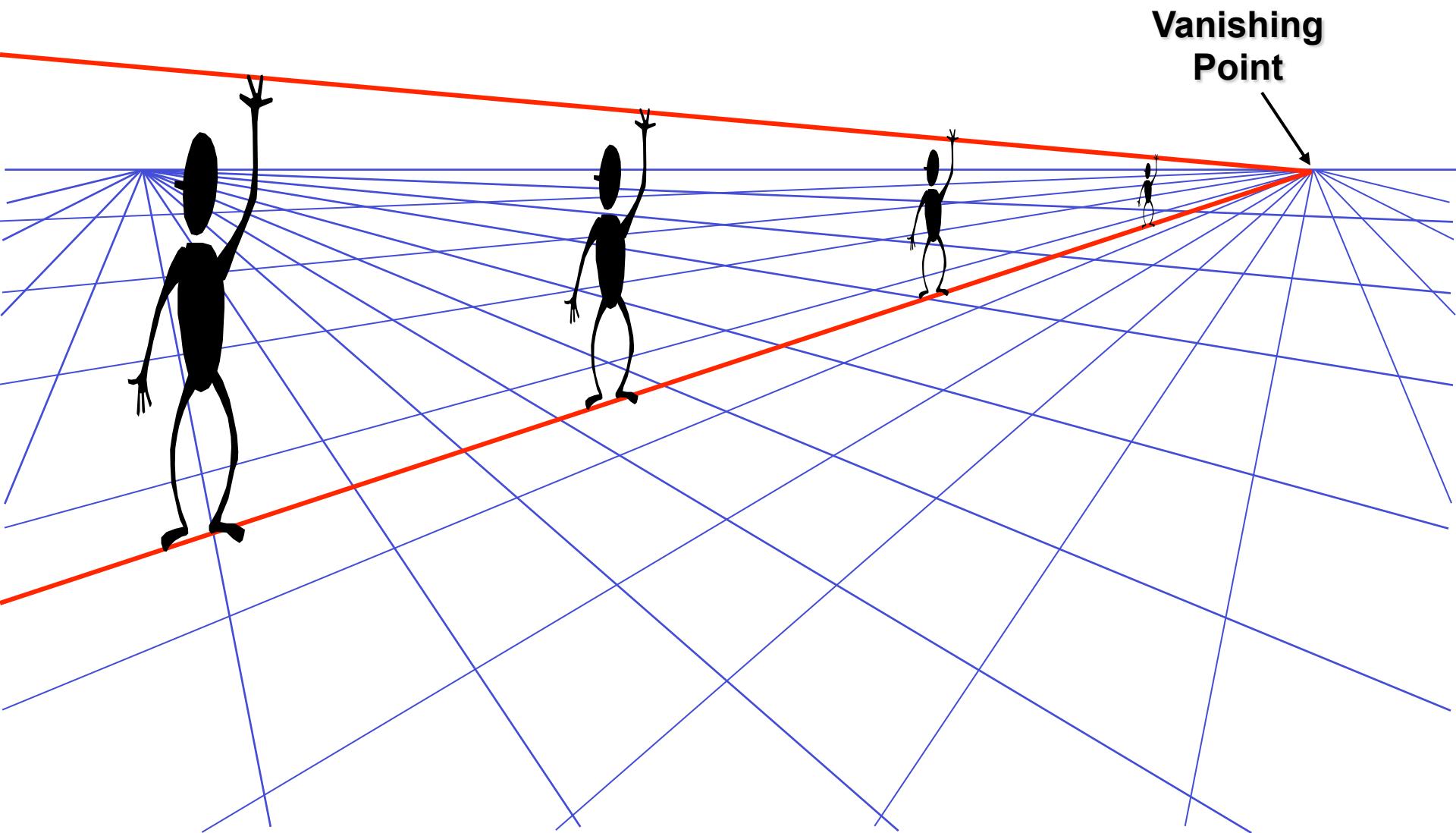
Making measurements from a single image



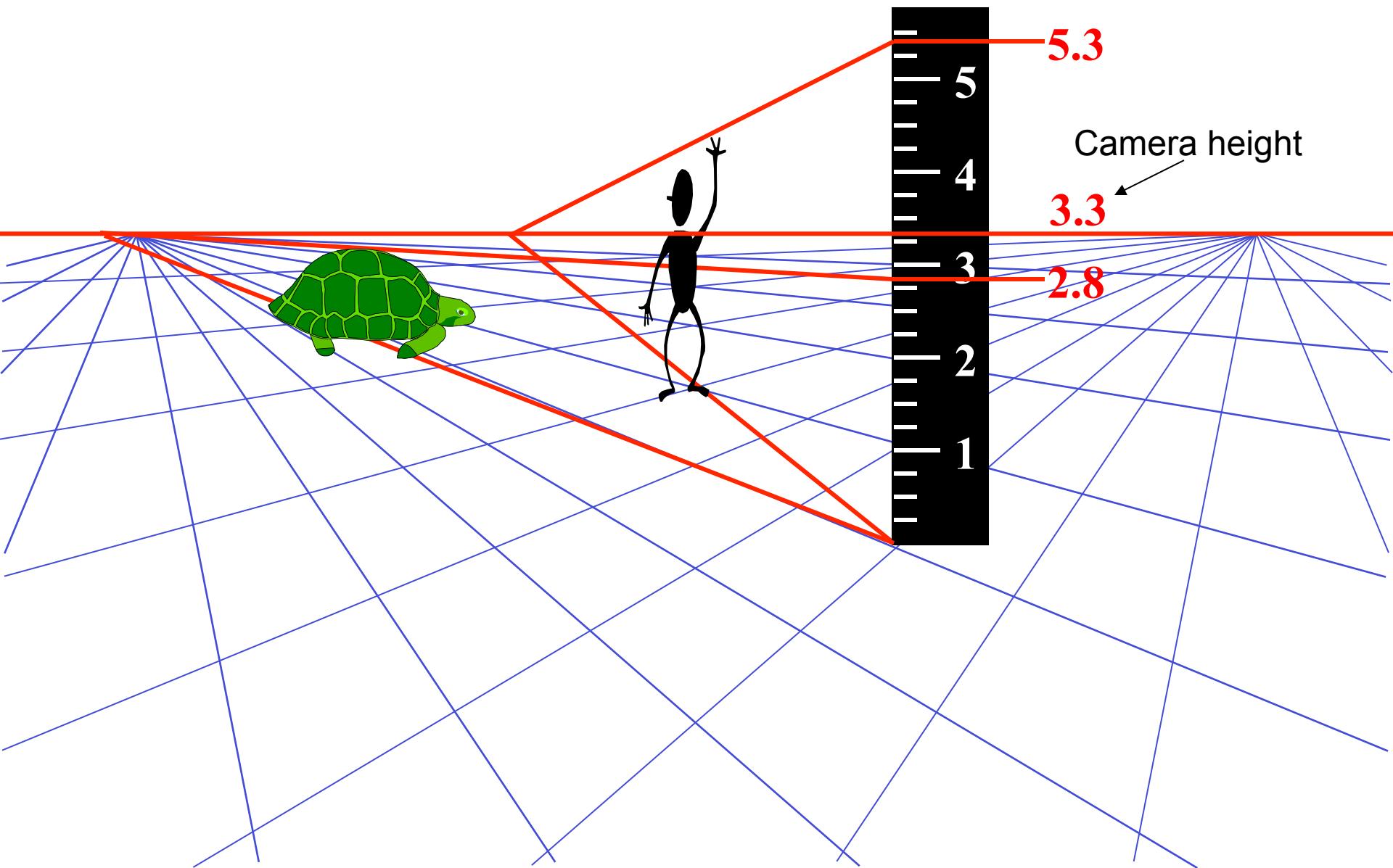
http://en.wikipedia.org/wiki/Ames_room

Source: S. Lazebnik

Comparing heights



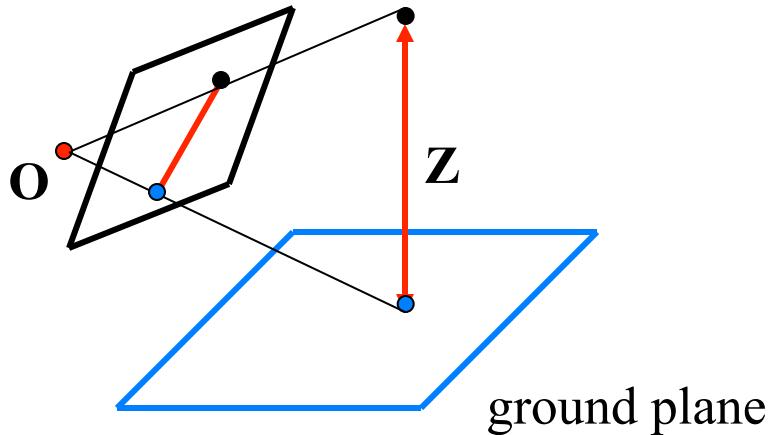
Measuring height



Which is higher – the camera or the man in the parachute?



Measuring height without a ruler



Compute Z from image measurements

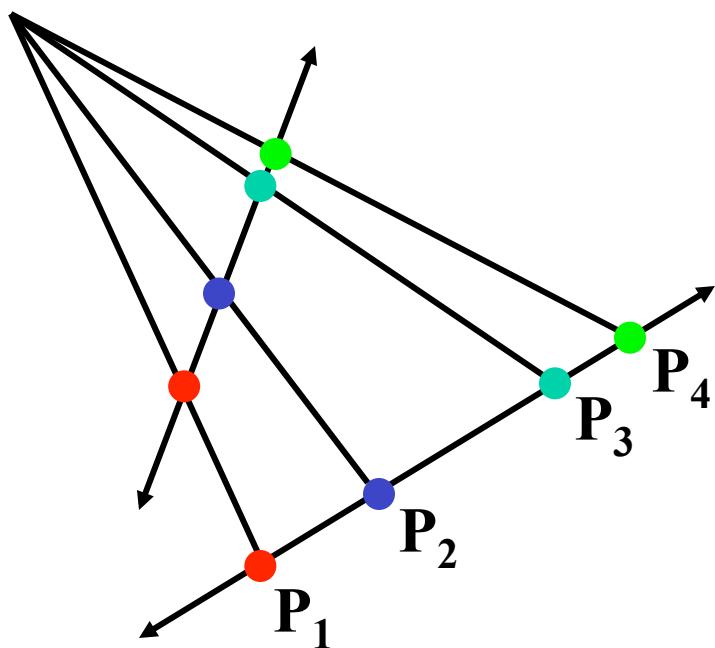
- Need more than vanishing points to do this

Projective invariant

- We need to use a *projective invariant*: a quantity that does not change under projective transformations (including perspective projection)
 - What are some invariants for similarity, affine transformations?

Projective invariant

- We need to use a *projective invariant*: a quantity that does not change under projective transformations (including perspective projection)
- The *cross-ratio* of four points:



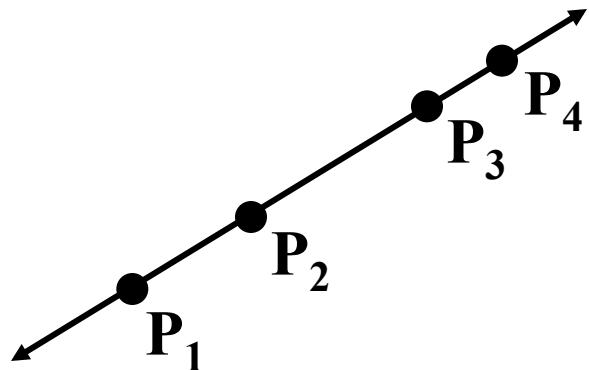
$$\frac{\|P_3 - P_1\| \|P_4 - P_2\|}{\|P_3 - P_2\| \|P_4 - P_1\|}$$

The cross ratio

A Projective Invariant

- Something that does not change under projective transformations (including perspective projection)

The cross-ratio of 4 collinear points



$$\frac{\|\mathbf{P}_3 - \mathbf{P}_1\| \|\mathbf{P}_4 - \mathbf{P}_2\|}{\|\mathbf{P}_3 - \mathbf{P}_2\| \|\mathbf{P}_4 - \mathbf{P}_1\|}$$

$$\mathbf{P}_i = \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

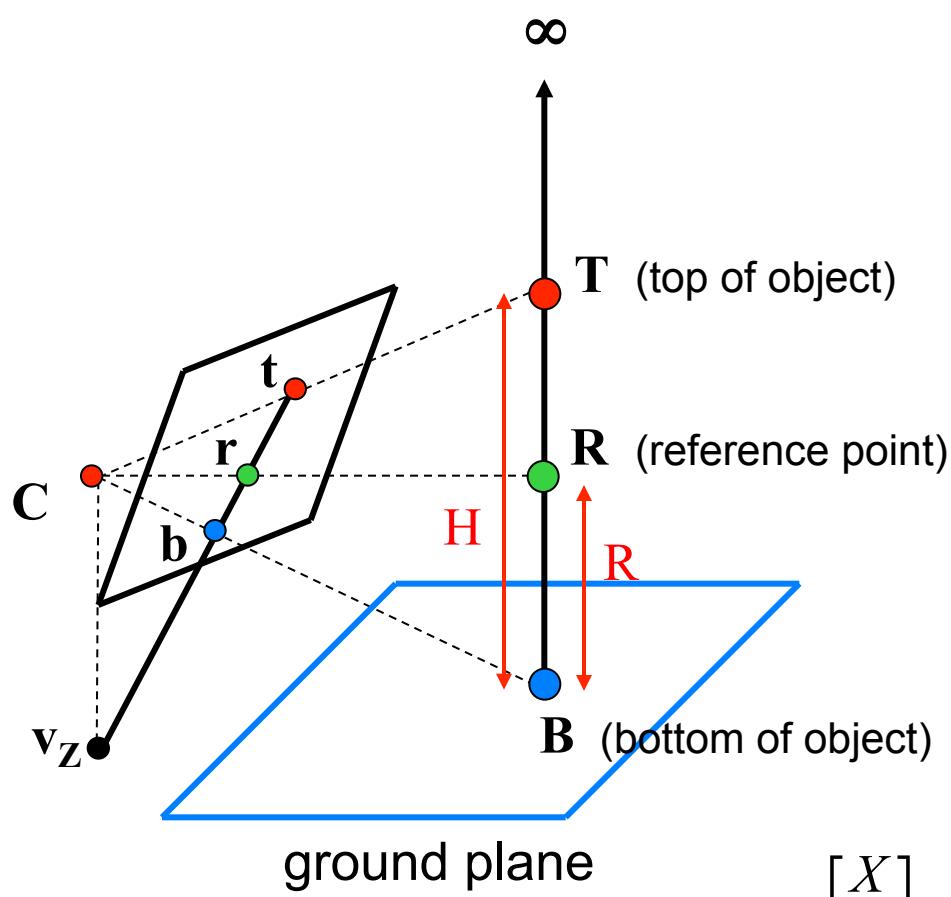
Can permute the point ordering

- $4! = 24$ different orders (but only 6 distinct values)

$$\frac{\|\mathbf{P}_1 - \mathbf{P}_3\| \|\mathbf{P}_4 - \mathbf{P}_2\|}{\|\mathbf{P}_1 - \mathbf{P}_2\| \|\mathbf{P}_4 - \mathbf{P}_3\|}$$

This is the fundamental invariant of projective geometry

Measuring height



scene points represented as $\mathbf{P} = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$

$$\frac{\|\mathbf{B} - \mathbf{T}\| \|\infty - \mathbf{R}\|}{\|\mathbf{B} - \mathbf{R}\| \|\infty - \mathbf{T}\|} = \frac{H}{R}$$

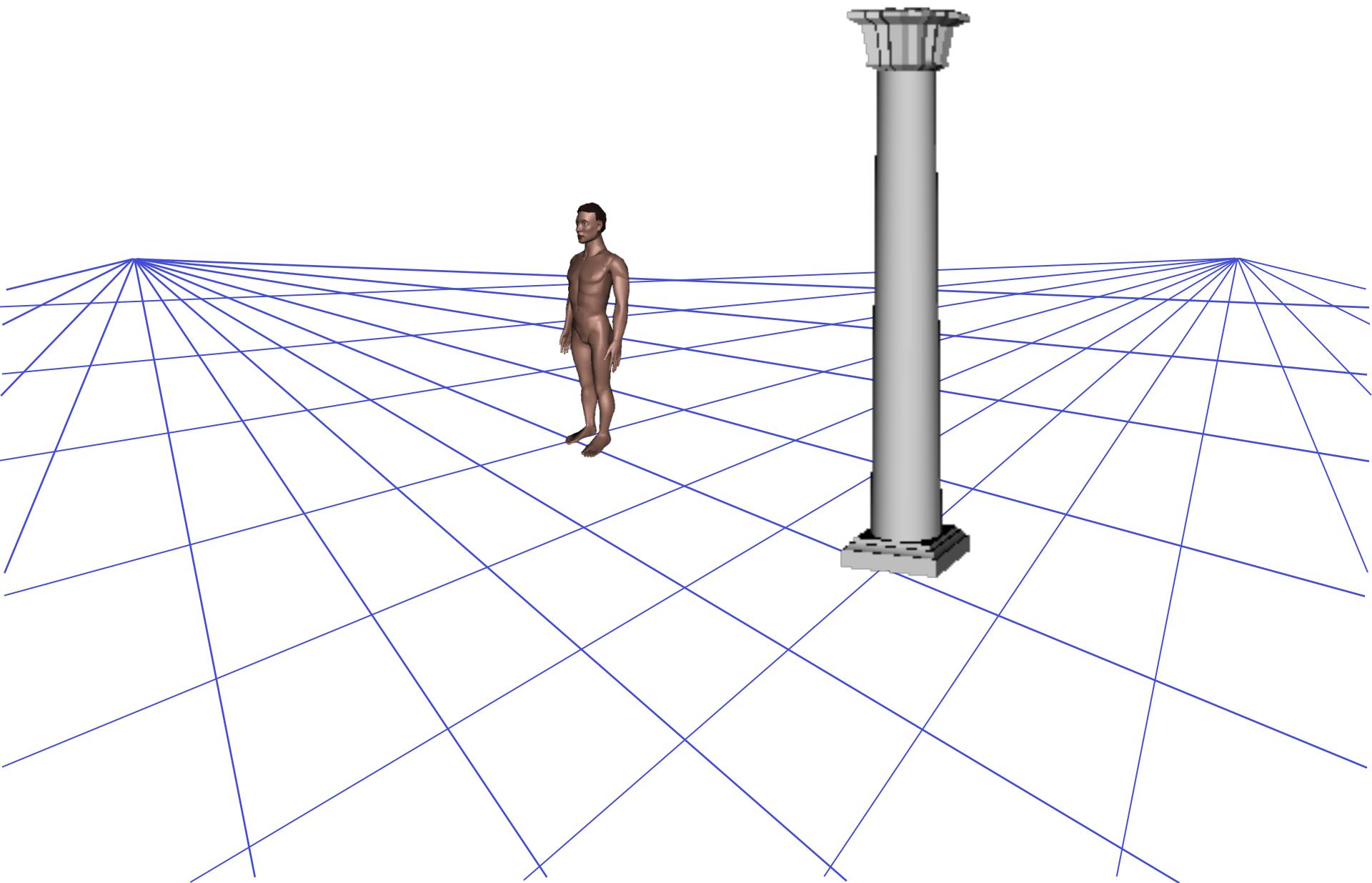
scene cross ratio

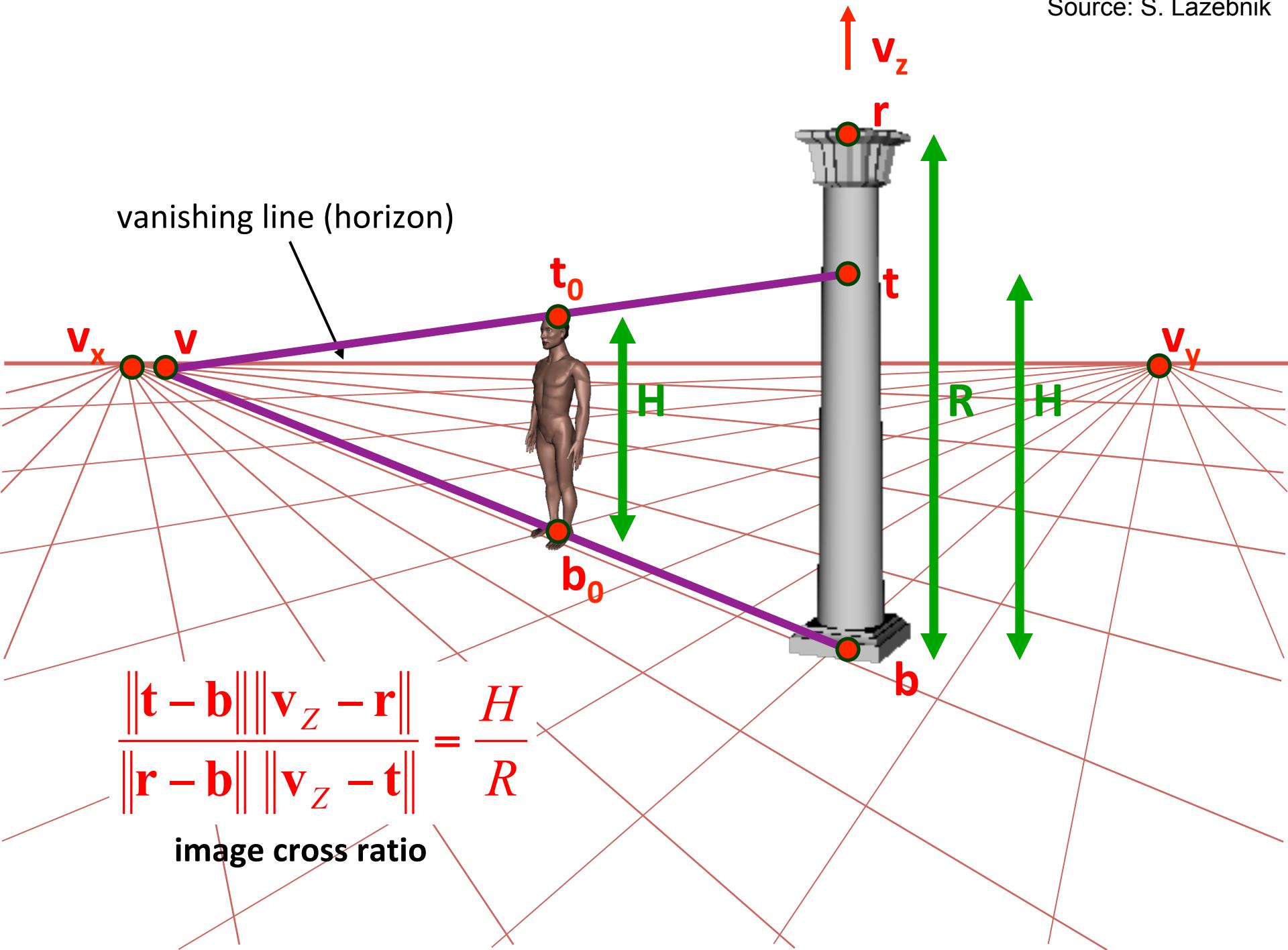
$$\frac{\|\mathbf{b} - \mathbf{t}\| \|\mathbf{v}_Z - \mathbf{r}\|}{\|\mathbf{b} - \mathbf{r}\| \|\mathbf{v}_Z - \mathbf{t}\|} = \frac{H}{R}$$

image cross ratio

image points as $\mathbf{p} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$

Measuring height without a ruler



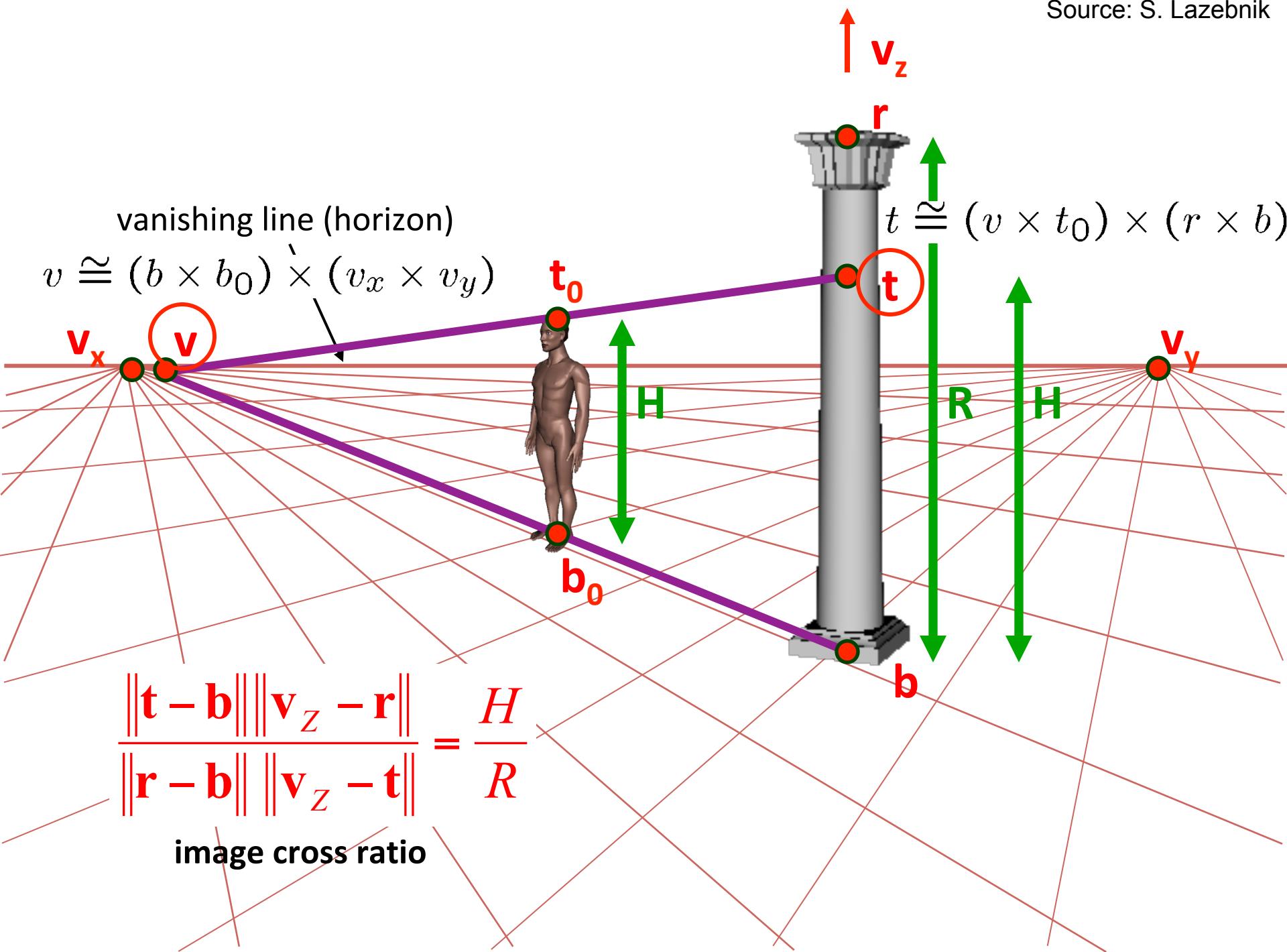


2D lines in homogeneous coordinates

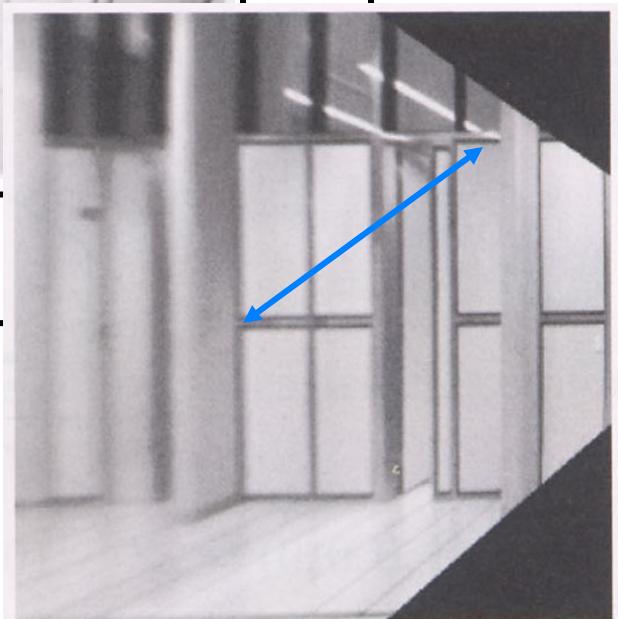
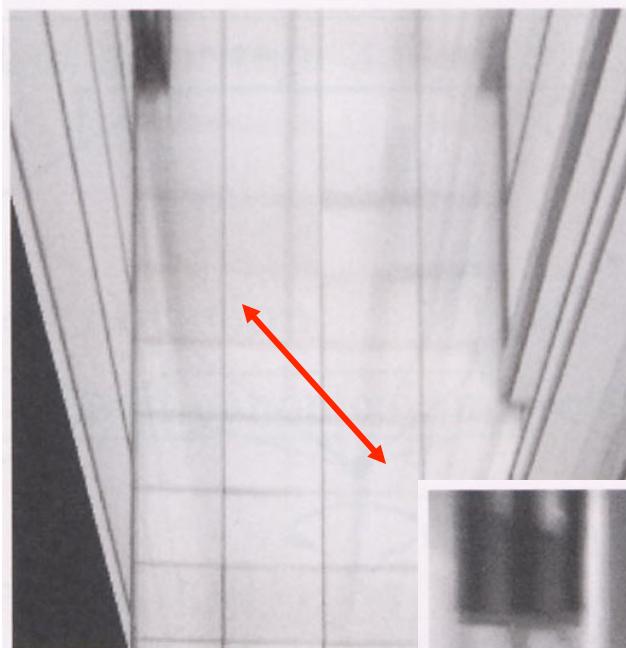
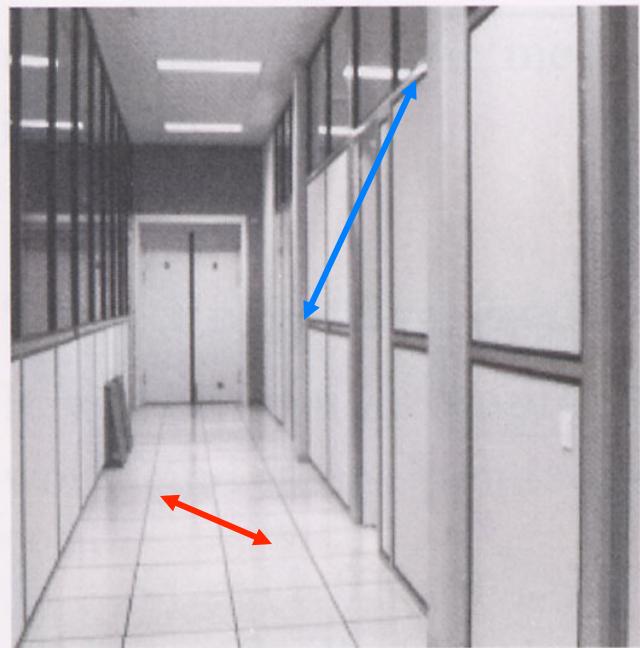
- Line equation: $ax + by + c = 0$

$$\mathbf{l}^T \mathbf{x} = 0 \quad \text{where} \quad \mathbf{l} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Line passing through two points: $\mathbf{l} = \mathbf{x}_1 \times \mathbf{x}_2$
- Intersection of two lines: $\mathbf{x} = \mathbf{l}_1 \times \mathbf{l}_2$
 - What is the intersection of two parallel lines?



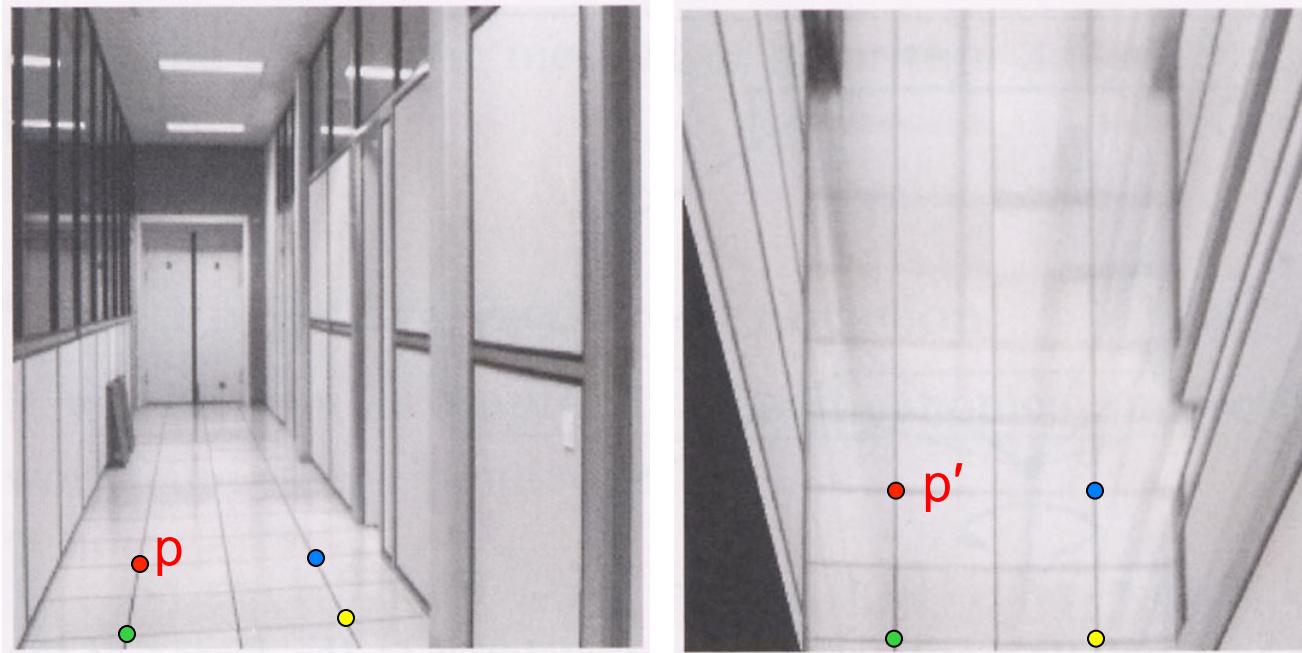
Measurements on planes



Approach: un warp then measure

What kind of warp is this?

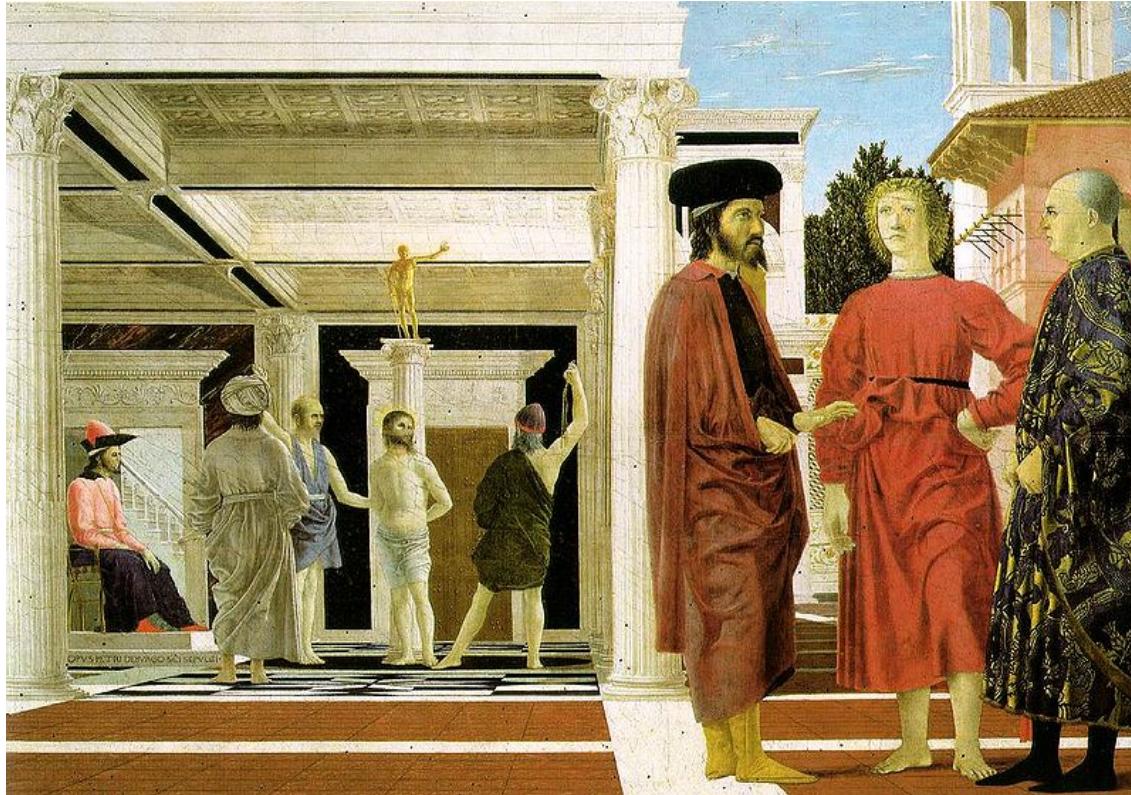
Image rectification



To un warp (rectify) an image

- solve for homography H given p and p'
 - how many points are necessary to solve for H ?

Image rectification: example

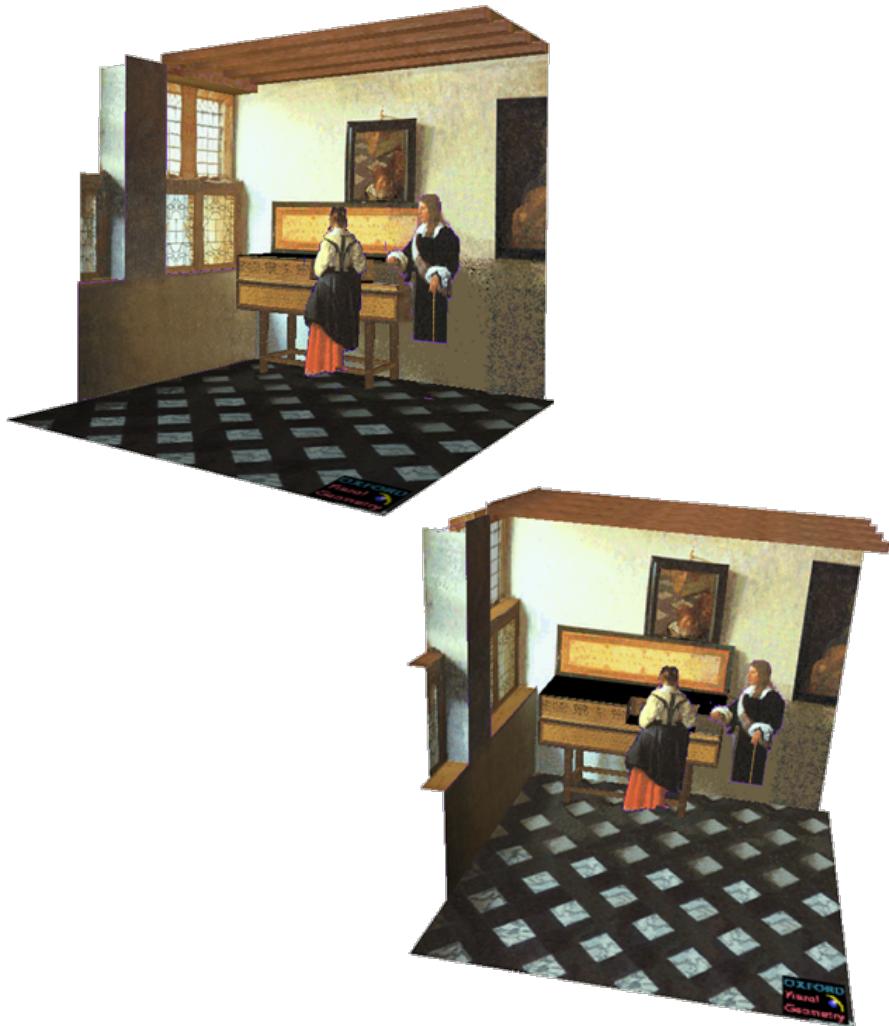


Piero della Francesca, *Flagellation*, ca. 1455

Application: 3D modeling from a single image



J. Vermeer, Music Lesson, 1662



A. Criminisi, M. Kemp, and A. Zisserman,
[Bringing Pictorial Space to Life: computer techniques for the analysis of paintings](#), Proc. Computers and the History of Art, 2002

http://research.microsoft.com/en-us/um/people/antcrim/ACriminisi_3D_Museum.wmv

Application: 3D modeling from a single image



D. Hoiem, A.A. Efros, and M. Hebert, "Automatic Photo Pop-up", SIGGRAPH 2005.

http://dhoiem.cs.illinois.edu/projects/popup/popup_movie_450_250.mp4

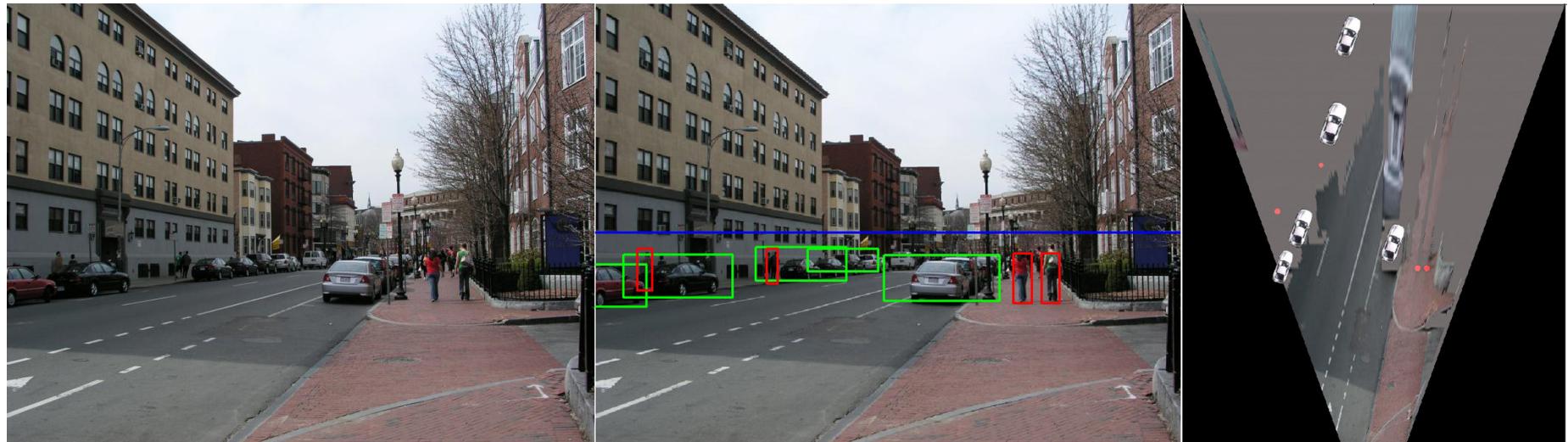
Application: Image editing

Inserting synthetic objects into images:
<http://vimeo.com/28962540>



K. Karsch and V. Hedau and D. Forsyth and D. Hoiem, “Rendering Synthetic Objects into Legacy Photographs,” *SIGGRAPH Asia 2011*

Application: Object recognition



D. Hoiem, A.A. Efros, and M. Hebert, "Putting Objects in Perspective", CVPR 2006