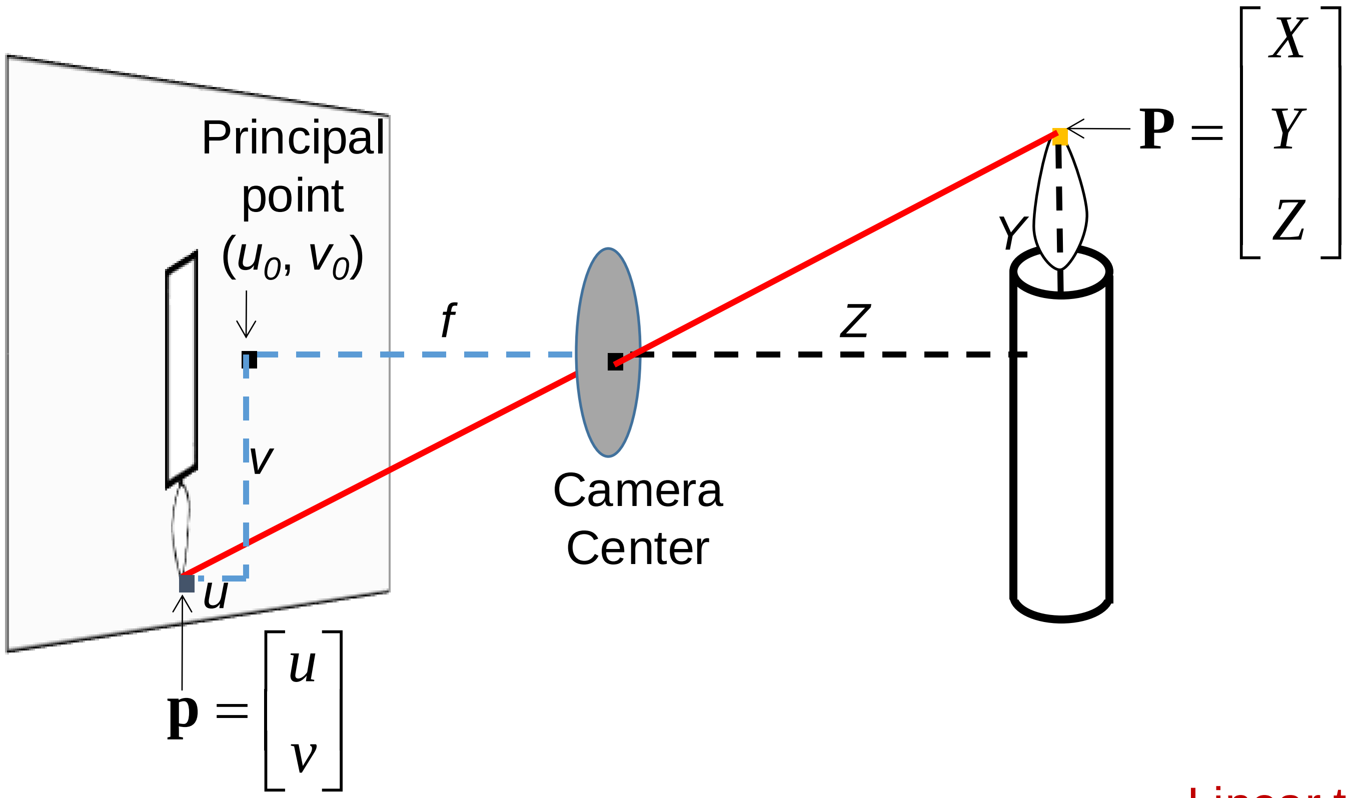
Single-View Geometry

Jiayuan Gu

gujy1@shanghaitech.edu.cn

Review: Perspective Projection



Principal point (u_0, v_0)
f
Z
Y
$\mathbf{P} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$
v
u
Camera Center
$\mathbf{p} = \begin{bmatrix} u \\ v \end{bmatrix}$

$$u = f\,\frac{x}{z}$$

$$v = f\,\frac{y}{z}$$

$$w \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Linear transform in homogeneous coordinates

Source: Derek Hoiem

Review: Homogeneous Coordinates

Converting **to** homogeneous coordinates

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

homogeneous image coordinates

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

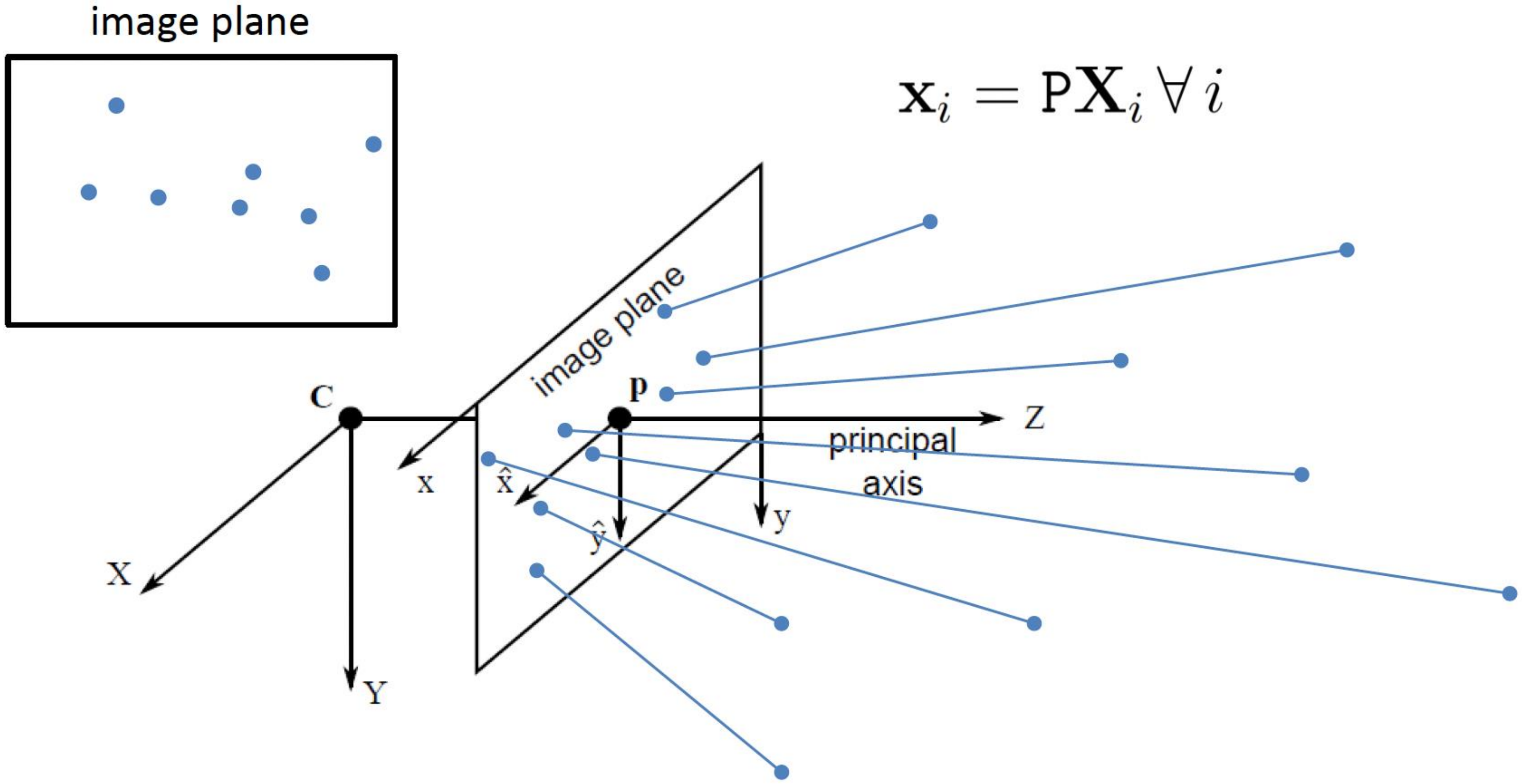
homogeneous scene coordinates

Converting **from** homogeneous coordinates

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

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

Review: Camera Projection Matrix

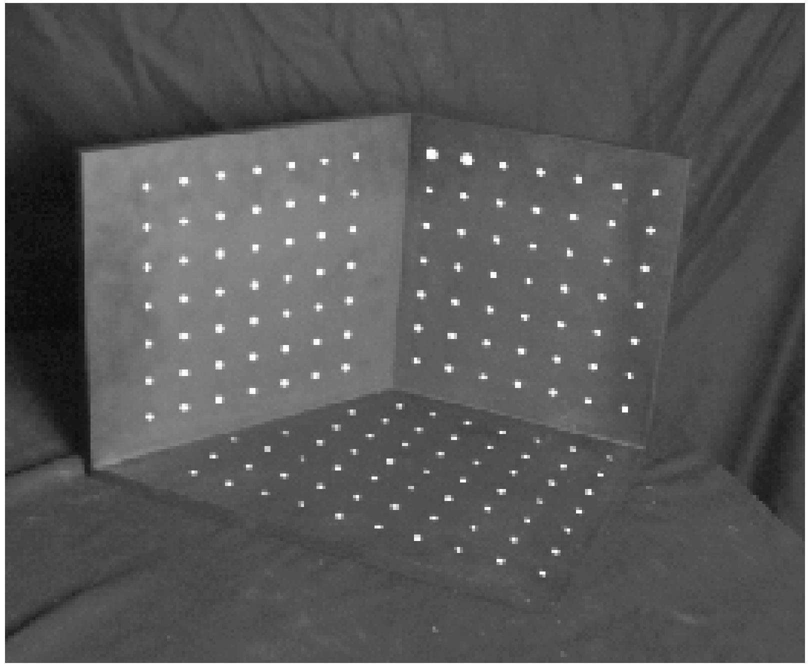


$$\mathbf{x}_i = \mathrm{P}\mathbf{X}_i \, \forall \, i$$

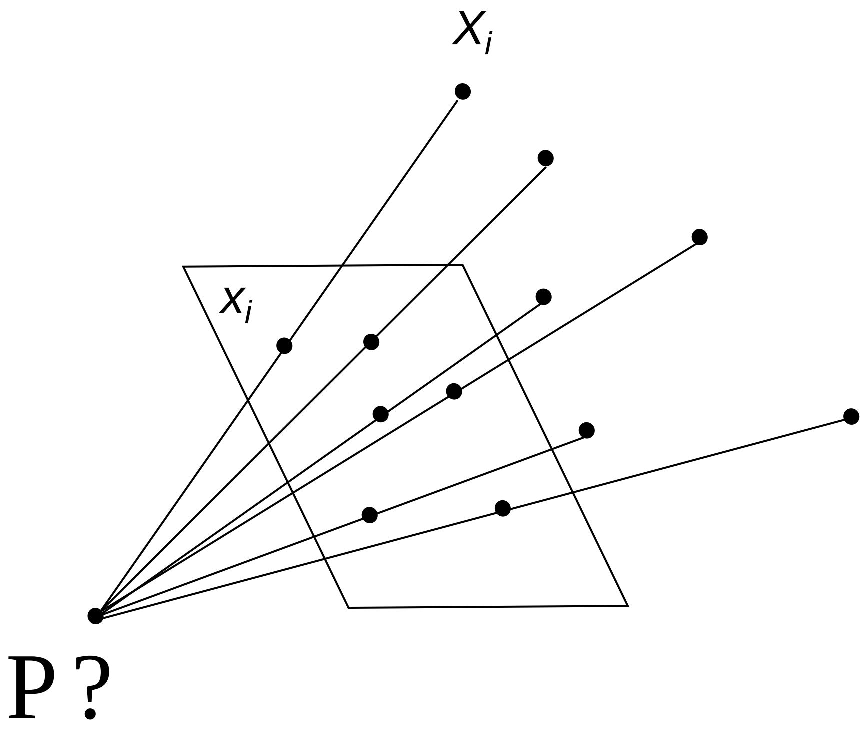
Source: Ben Ocha’s CSE272-WI24

Review: Camera Calibration from 2D-3D Pairs

- Given n points with known 3D coordinates X_i and known image projections x_i, estimate the camera parameters



A calibration grid with known geometry

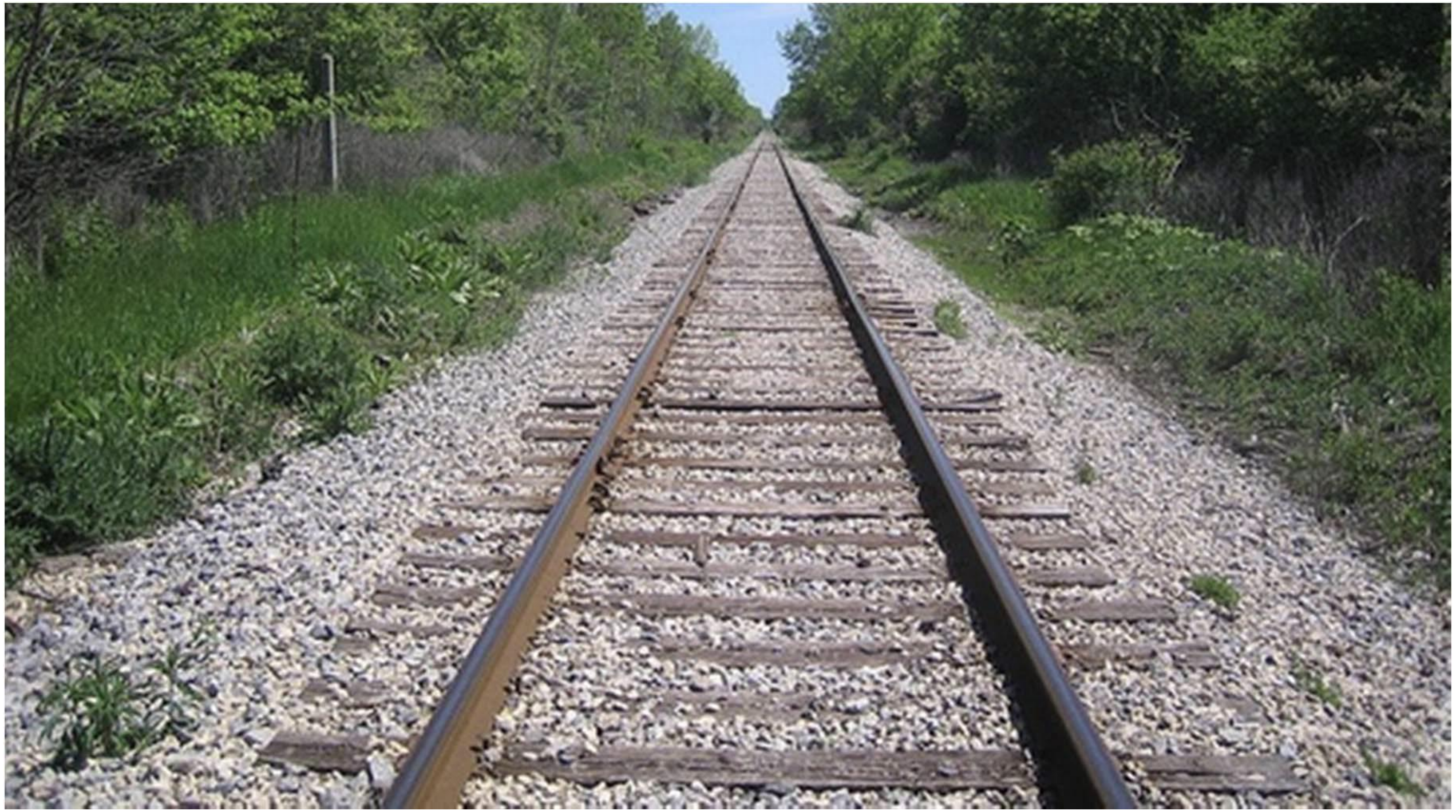


Today’s Topic

- Can we calibrate the camera without 2D-3D pairs?
    - With vanishing points

- How can we measure the object size in the world from an image?

Properties of Perspective Projection

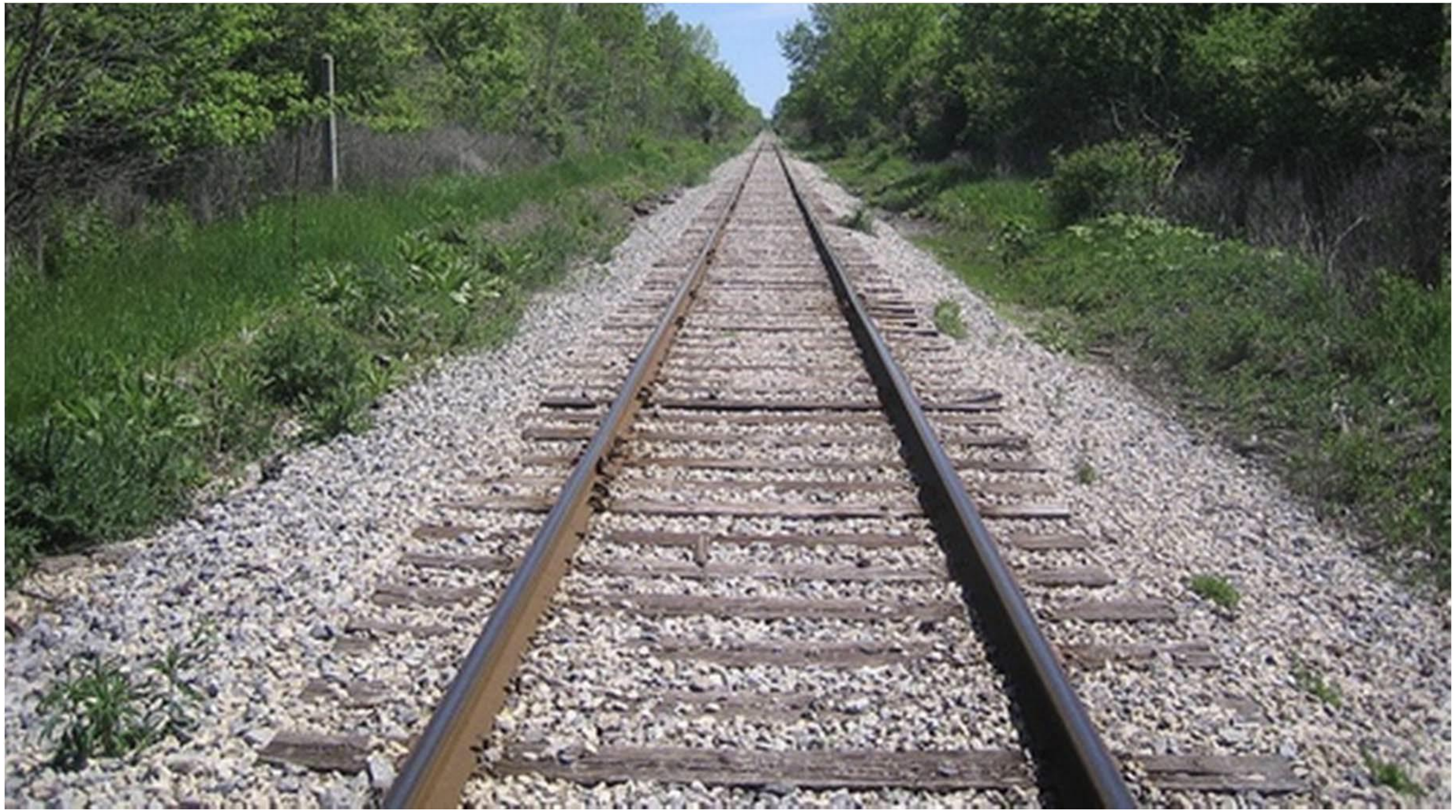


Points projected to points

Lines projected to lines

Source: Stanford CS231a Lec8

Properties of Perspective Projection

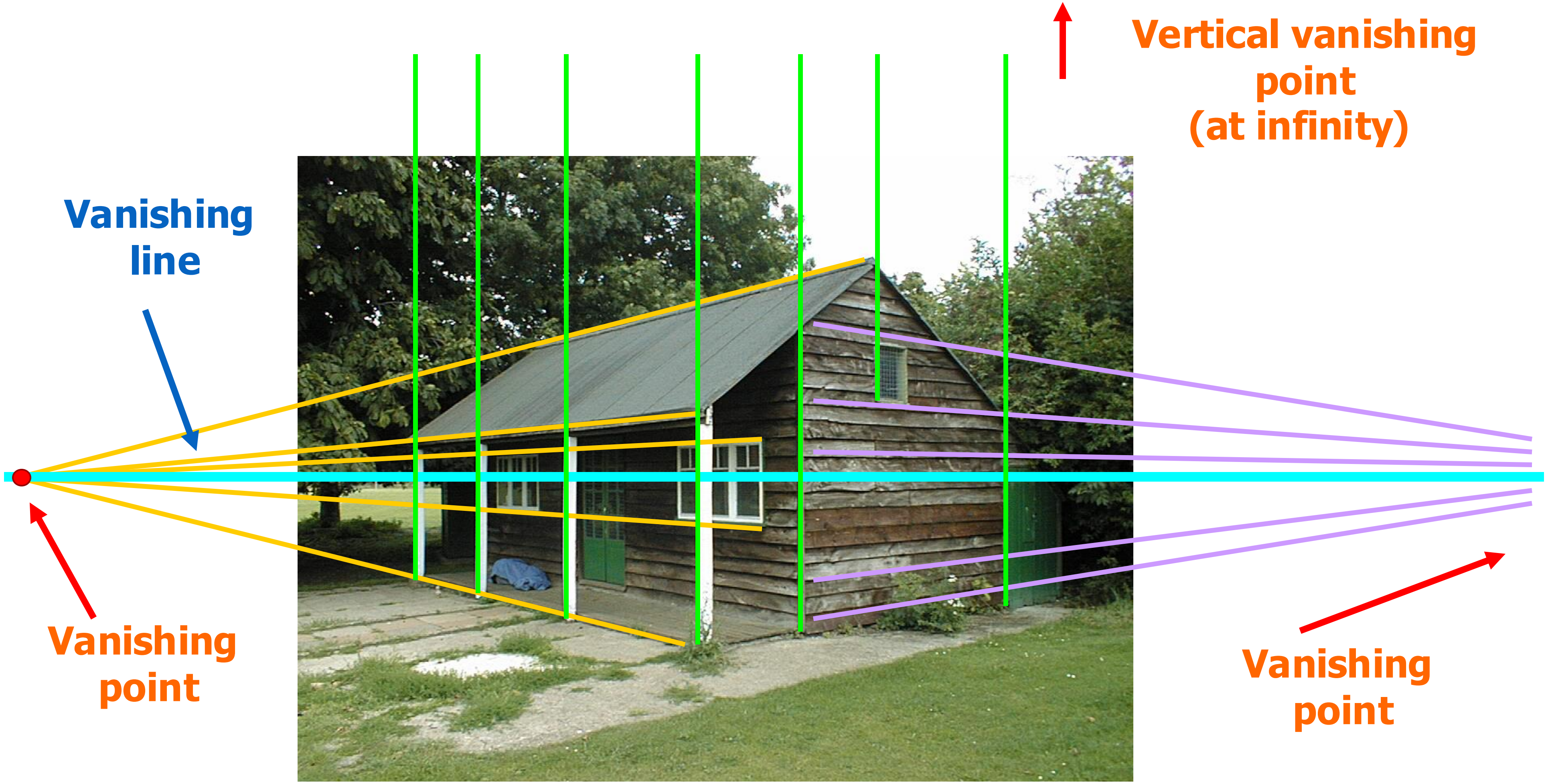


Angles are not preserved

Parallel lines meet

Source: Stanford CS231a Lec8

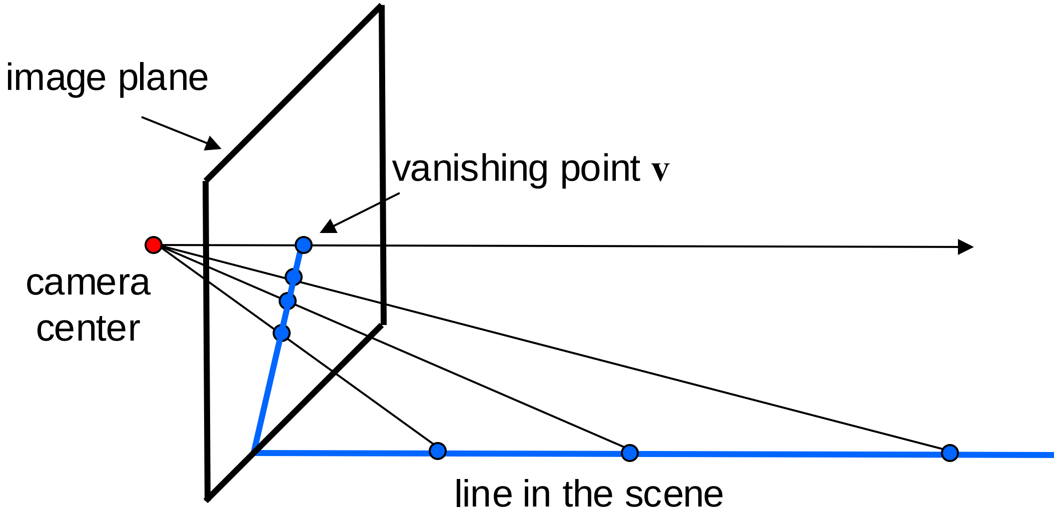
Vanishing Points and Lines



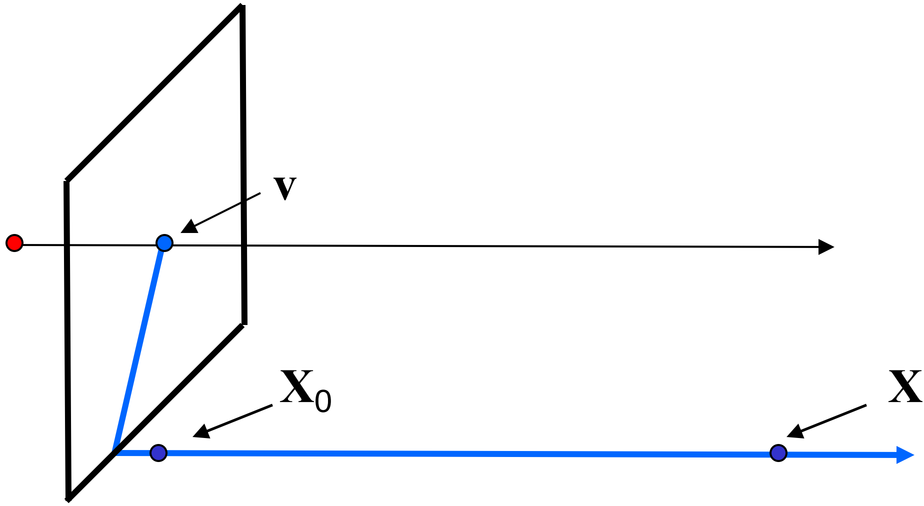
Slide from Efros, Photo from Criminisi

Vanishing Point

- Projections of parallel 3D lines intersect at a vanishing point
- Not all lines the projections of which intersect are parallel
- Vanishing point <-> 3D direction of a line



Computing Vanishing Points



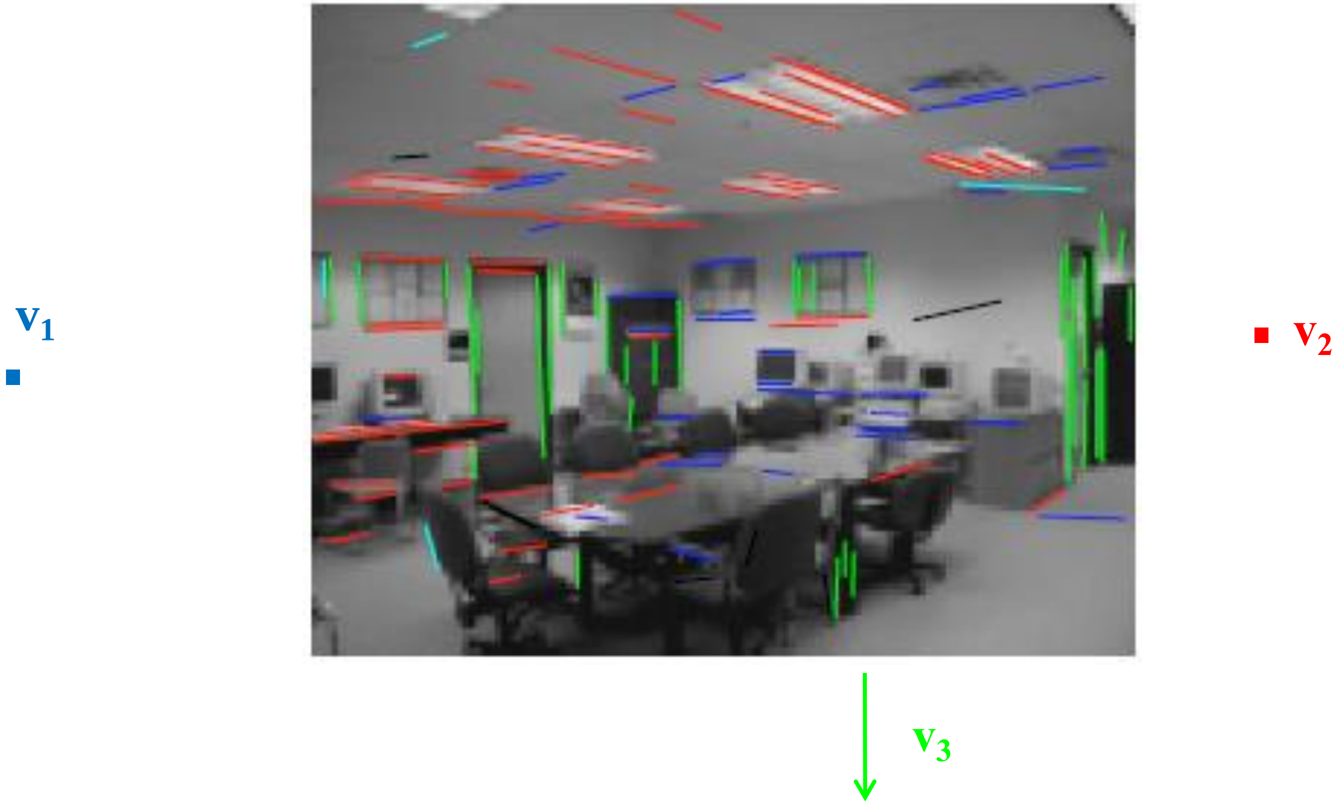
$\mathbf{X}_\infty$ is a *point at infinity,* $\mathbf{v}$ is its projection: $\mathbf{v} = \mathbf{P}\mathbf{X}_\infty$
The vanishing point depends only on *line direction*
All lines having direction $\mathbf{d}$ intersect at $\mathbf{X}_\infty$

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

$$\mathbf{X}_\infty = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ 0 \end{bmatrix}$$

Calibration from Vanishing Points

Consider a scene with three orthogonal vanishing directions:



Note: $\mathbf{v}_1$, $\mathbf{v}_2$ are *finite* vanishing points and $\mathbf{v}_3$ is an *infinite* vanishing point

Calibration from Vanishing Points

$$\mathbf{P}X = \begin{bmatrix} \boxed{\begin{matrix} * \\ * \\ * \end{matrix}} & \boxed{\begin{matrix} * \\ * \\ * \end{matrix}} & \boxed{\begin{matrix} * \\ * \\ * \end{matrix}} & \boxed{\begin{matrix} * \\ * \\ * \end{matrix}} \end{bmatrix} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 & \mathbf{p}_4 \end{bmatrix}$$

- $\mathbf{p}_1 = \mathbf{P}(1,0,0,0)^T$ – 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:

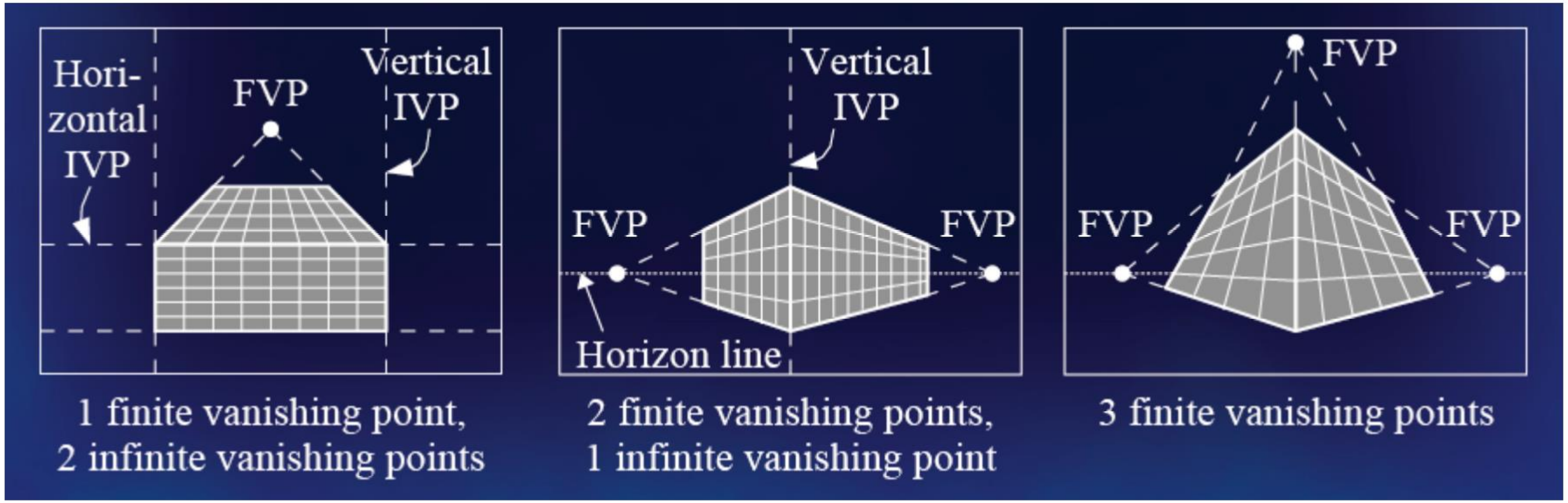
$$\mathbf{e_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e_2} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e_3} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \lambda_i \mathbf{v}_i = \mathbf{K}\left[\mathbf{R} \mid \mathbf{t}\right]\begin{bmatrix} \mathbf{e}_i \\ 0 \end{bmatrix} = \mathbf{K}\mathbf{R}\mathbf{e}_i$$

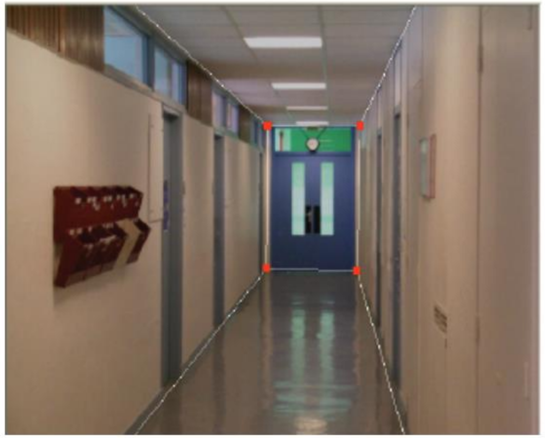
$$\mathbf{e}_i = \lambda_i \mathbf{R}^T \mathbf{K}^{-1} \mathbf{v}_i, \quad \mathbf{e}_i^T \mathbf{e}_j = 0$$

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

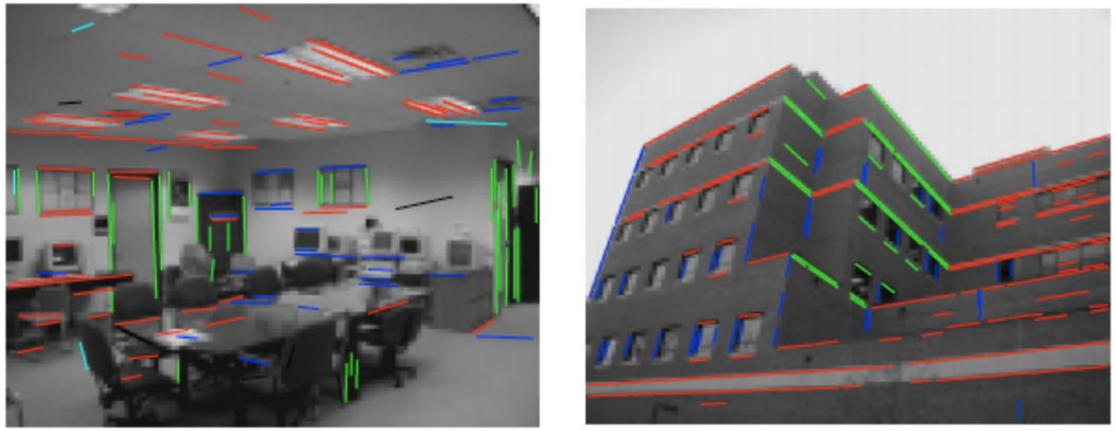
Each pair of vanishing points gives us a constraint on the focal length and principal point

Calibration from vanishing points


Hori-zontal IVP
FVP
Vertical IVP
1 finite vanishing point,
2 infinite vanishing points
Vertical IVP
FVP
FVP
Horizon line
2 finite vanishing points,
1 infinite vanishing point
FVP
FVP
FVP
3 finite vanishing points



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



Can solve for focal length, principal point

Source: ECE549

Rotation from Vanishing Points

K has been solved

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

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

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

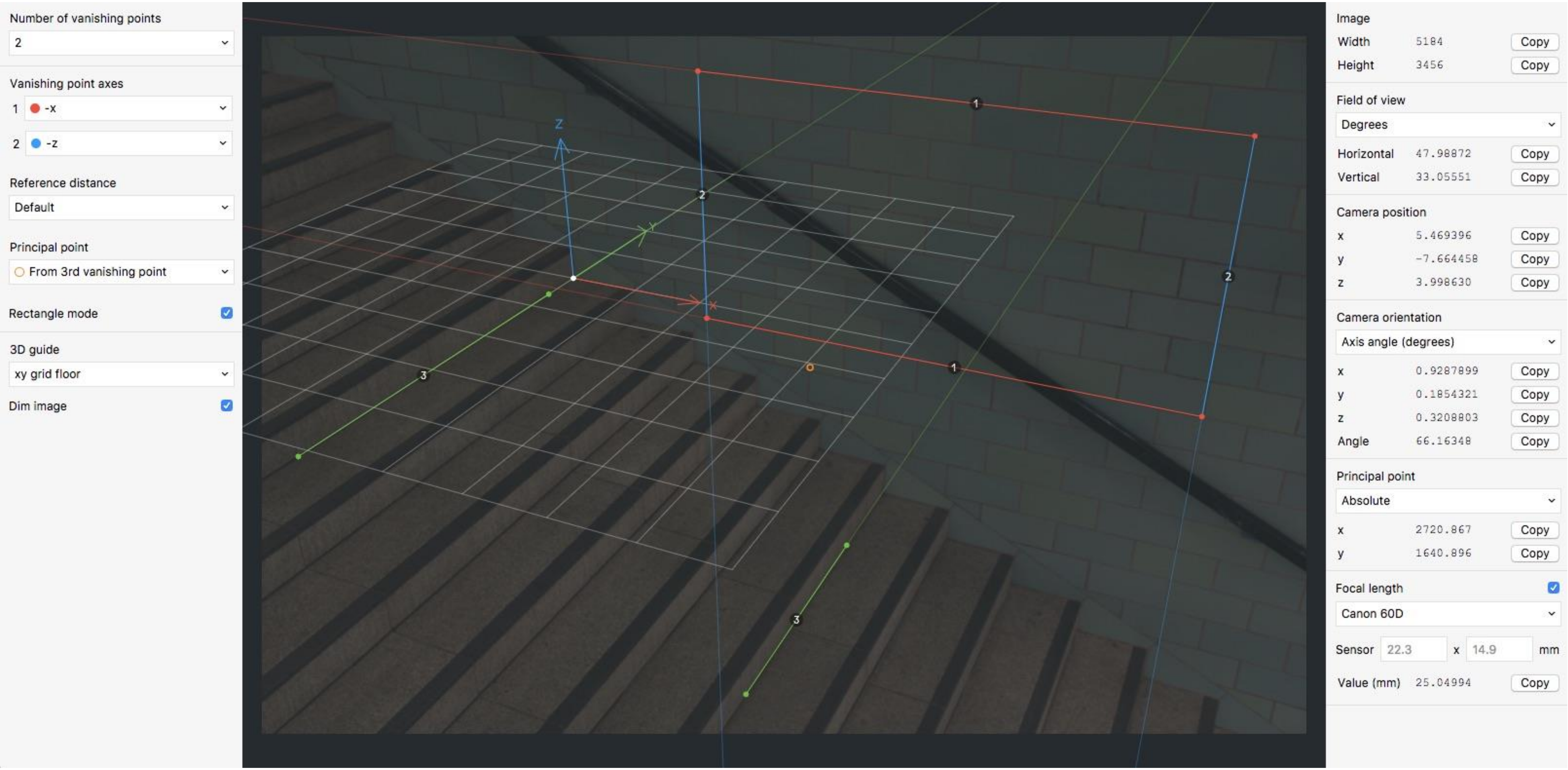
Get λ_i by using the constraint $||\mathbf{r}_i||^2=1$.

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

fSpy



Number of vanishing points: 2

Vanishing point axes
1 ● -x
2 ● -z

Reference distance: Default

Principal point: ○ From 3rd vanishing point

Rectangle mode ☑

3D guide: xy grid floor

Dim image ☑

Image
Width 5184
Height 3456

Field of view: Degrees
Horizontal 47.98872
Vertical 33.05551

Camera position
x 5.469396
y -7.664458
z 3.998630

Camera orientation: Axis angle (degrees)
x 0.9287899
y 0.1854321
z 0.3208803
Angle 66.16348

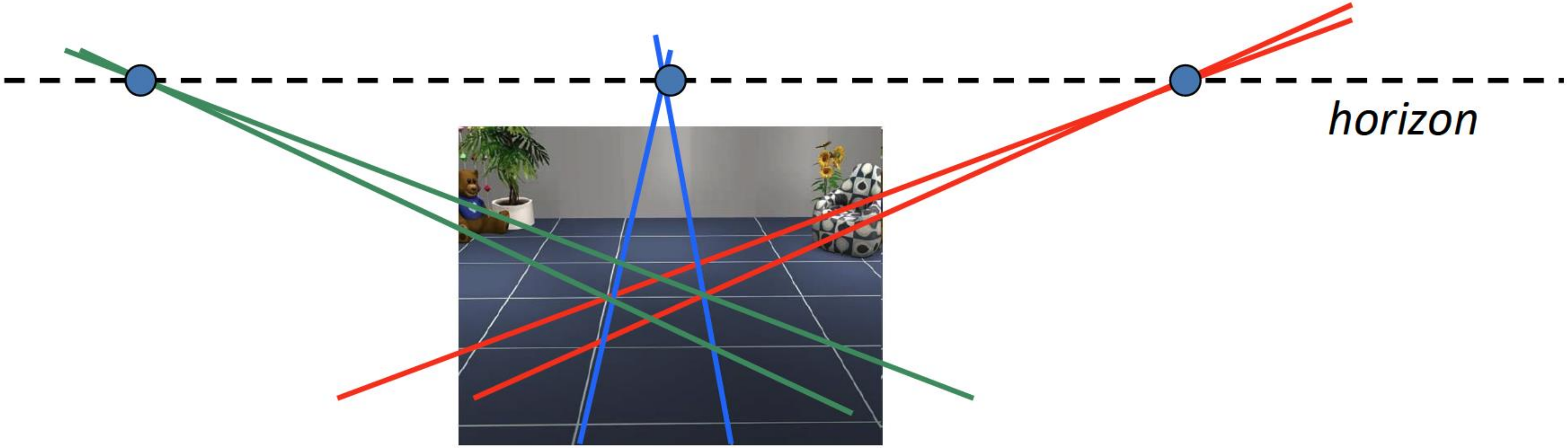
Principal point: Absolute
x 2720.867
y 1640.896

Focal length ☑
Canon 60D
Sensor 22.3 x 14.9 mm
Value (mm) 25.04994

https://github.com/stuffmatic/fSpy

Vanishing Line

- Projections of parallel 3D planes intersect at a **vanishing line**
- If a set of parallel 3D lines are also parallel to a particular plane, their vanishing point will lie on the vanishing line of the plane
- Vanishing line <-> 3D orientation of a plane



Source: Stanford CS231a Lec8

Vanishing Line

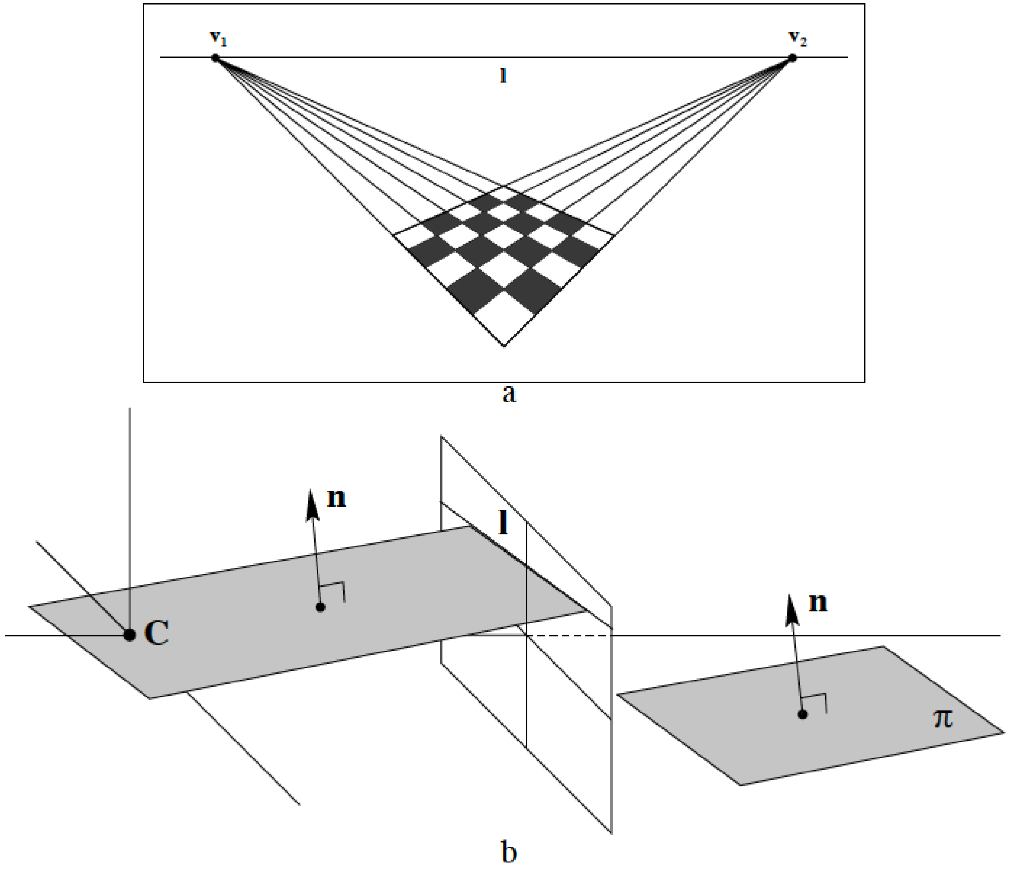
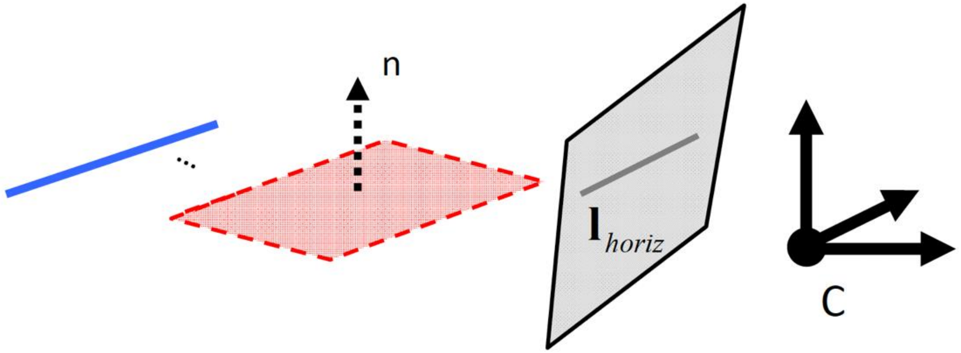
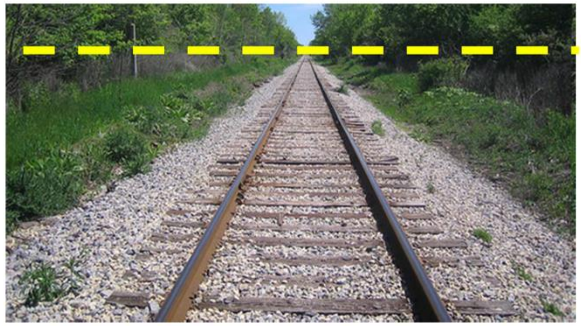


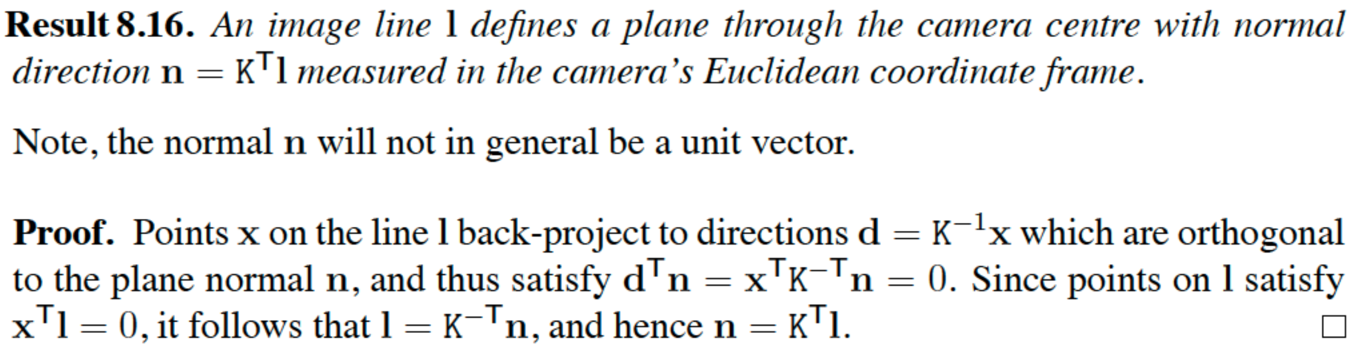
Fig. 8.16. **Vanishing line formation.** *(a) The two sets of parallel lines on the scene plane converge to the vanishing points $\mathbf{v}_1$ and $\mathbf{v}_2$ in the image. The line $\mathbf{l}$ through $\mathbf{v}_1$ and $\mathbf{v}_2$ is the vanishing line of the plane. (b) The vanishing line $\mathbf{l}$ of a plane π is obtained by intersecting the image plane with a plane through the camera centre $\mathbf{C}$ and parallel to π.*

Vanishing Line





$$\mathbf{n} = \mathrm{K}^{\mathrm{T}}\mathbf{l}_{\mathrm{horiz}}$$



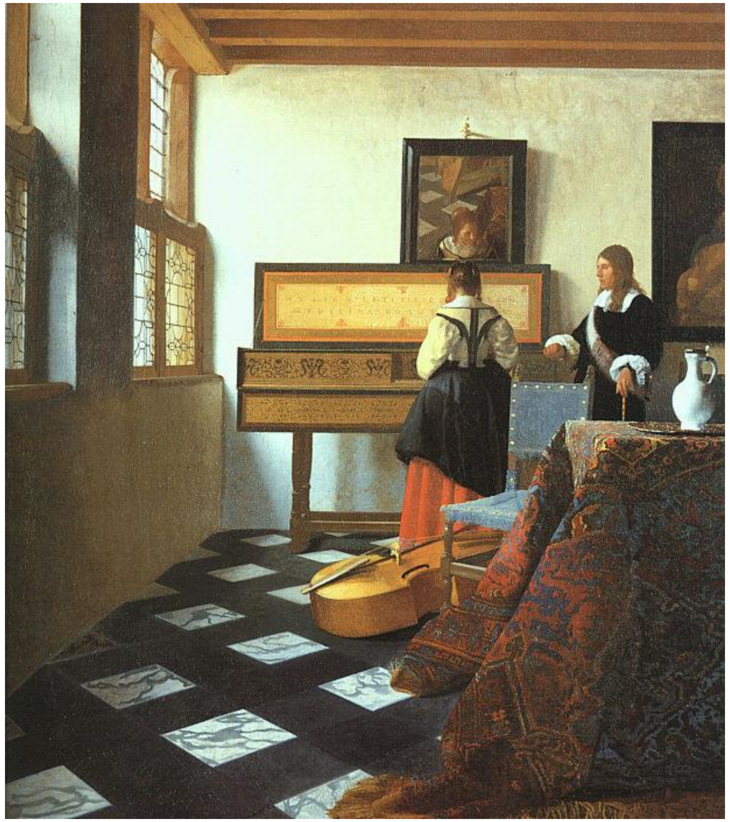
Result 8.16. *An image line* $\mathbf{l}$ *defines a plane through the camera centre with normal direction* $\mathbf{n} = \mathtt{K}^{\mathsf{T}}\mathbf{l}$ *measured in the camera's Euclidean coordinate frame.*

Note, the normal $\mathbf{n}$ will not in general be a unit vector.

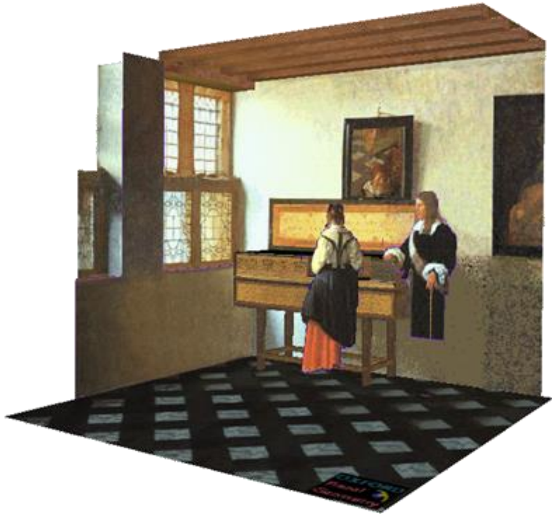
Proof. Points $\mathbf{x}$ on the line $\mathbf{l}$ back-project to directions $\mathbf{d} = \mathtt{K}^{-1}\mathbf{x}$ which are orthogonal to the plane normal $\mathbf{n}$, and thus satisfy $\mathbf{d}^{\mathsf{T}}\mathbf{n} = \mathbf{x}^{\mathsf{T}}\mathtt{K}^{-\mathsf{T}}\mathbf{n} = 0$. Since points on $\mathbf{l}$ satisfy $\mathbf{x}^{\mathsf{T}}\mathbf{l} = 0$, it follows that $\mathbf{l} = \mathtt{K}^{-\mathsf{T}}\mathbf{n}$, and hence $\mathbf{n} = \mathtt{K}^{\mathsf{T}}\mathbf{l}$. □

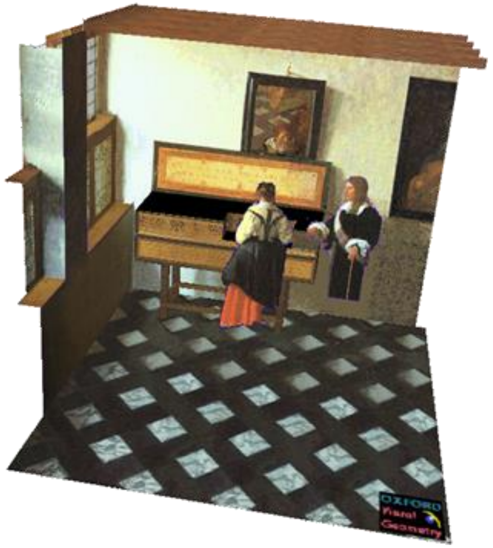
Source: Stanford CS231a Lec8

Application: 3D Modeling from a Single Image



J. Vermeer, *Music Lesson*, 1662



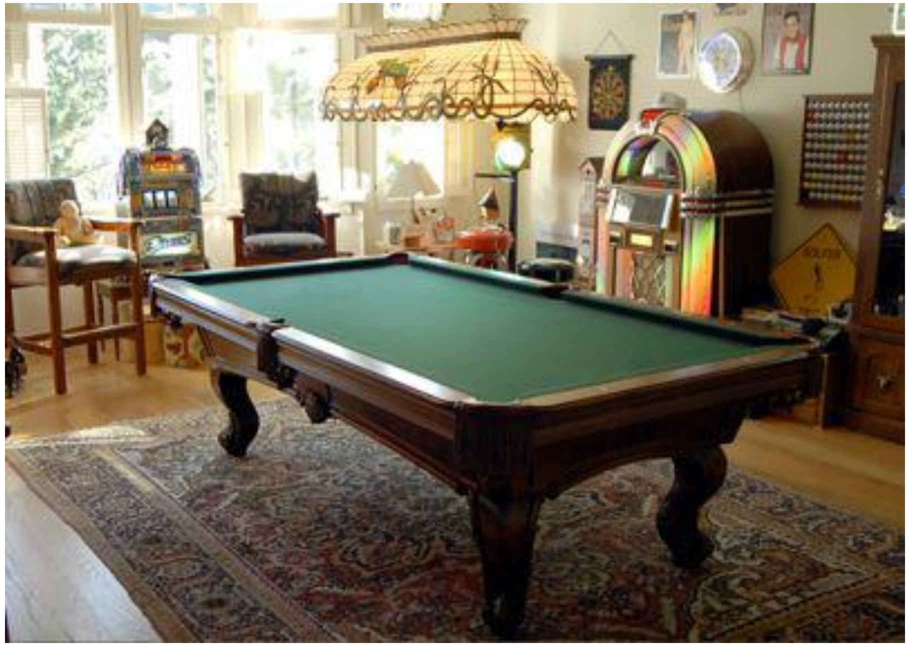


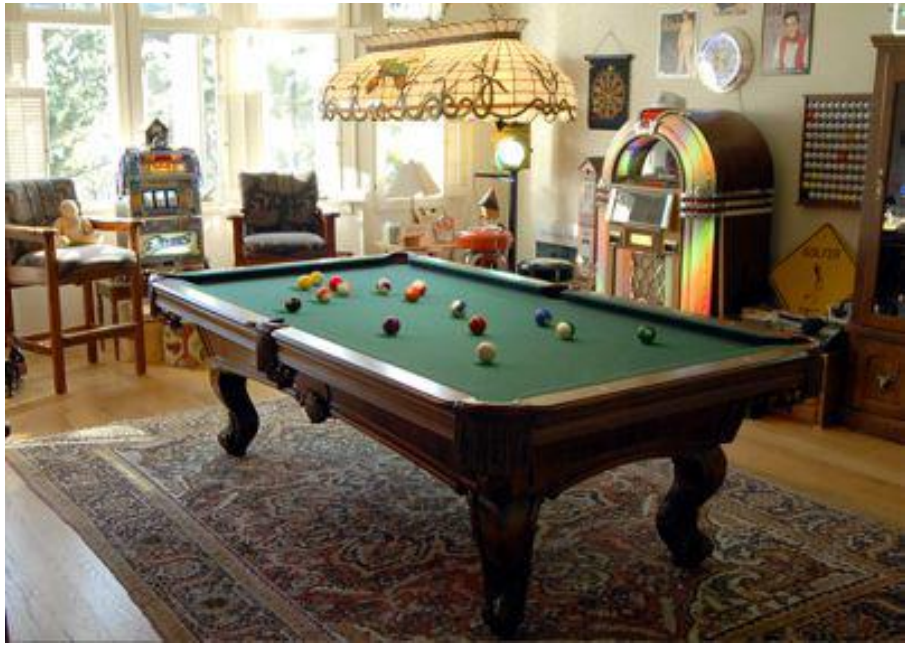
A. Criminisi, M. Kemp, and A. Zisserman, Bringing Pictorial Space to Life: computer techniques for the analysis of paintings, *Proc. Computers and the History of Art*, 2002

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

Application: Image editing

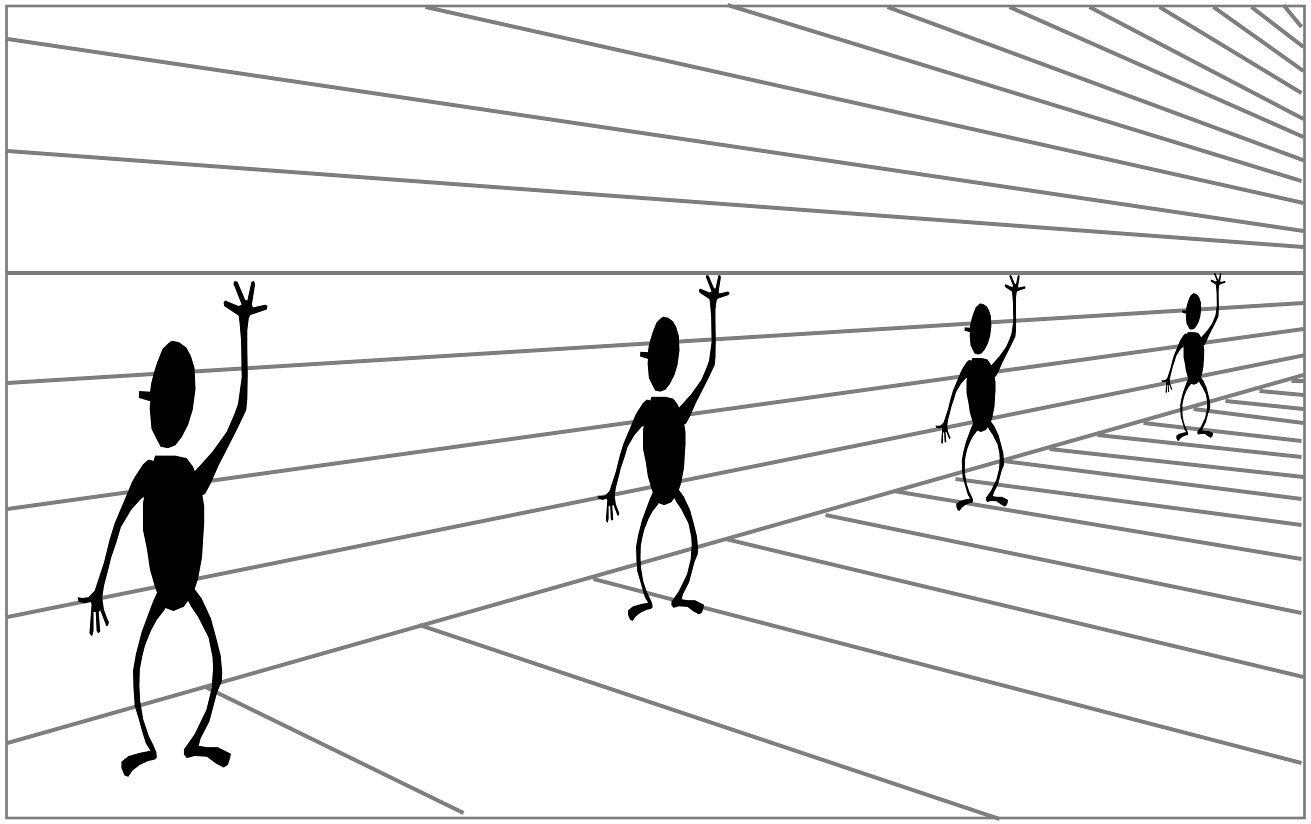
Inserting synthetic objects into images: http://vimeo.com/28962540





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

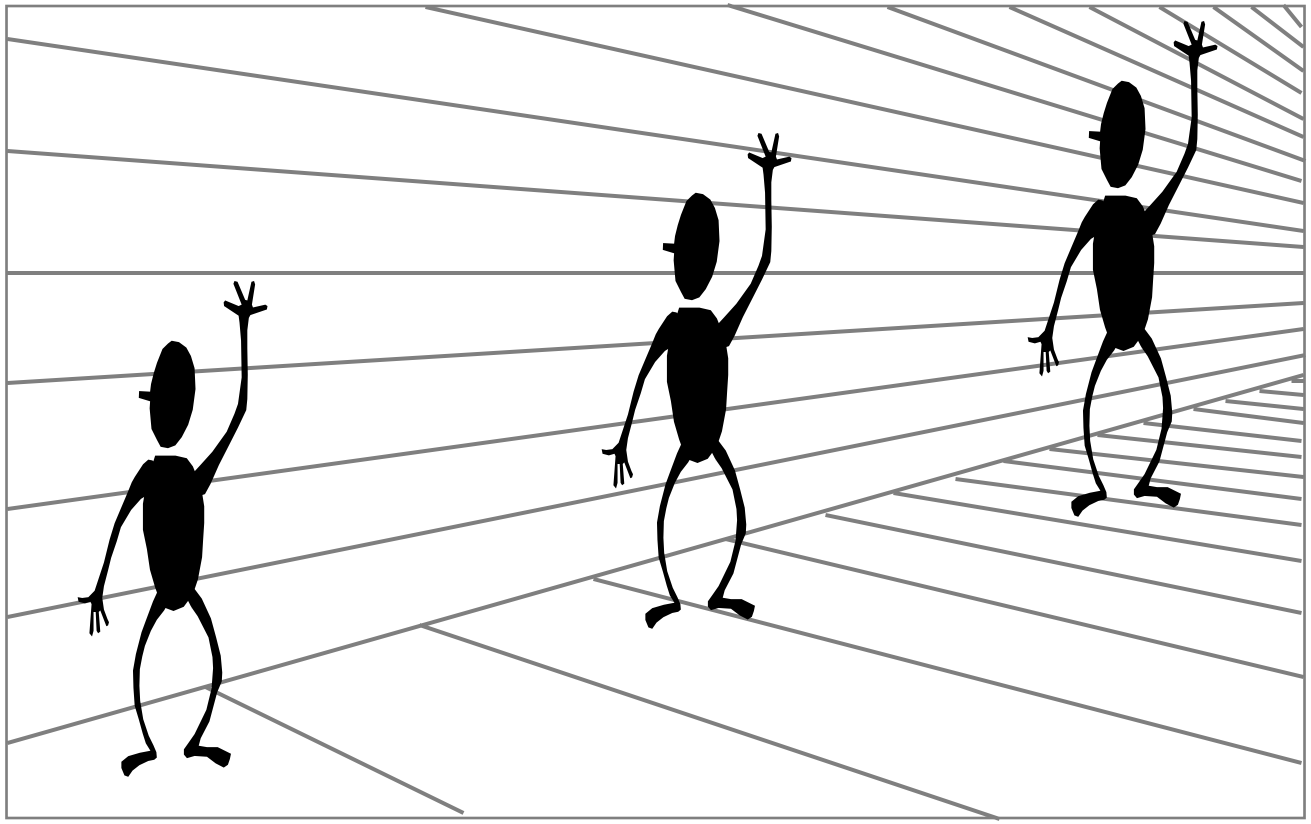
Estimating the 3D Size from a Single Image



Are they of the same height?

Slide by Steve Seitz

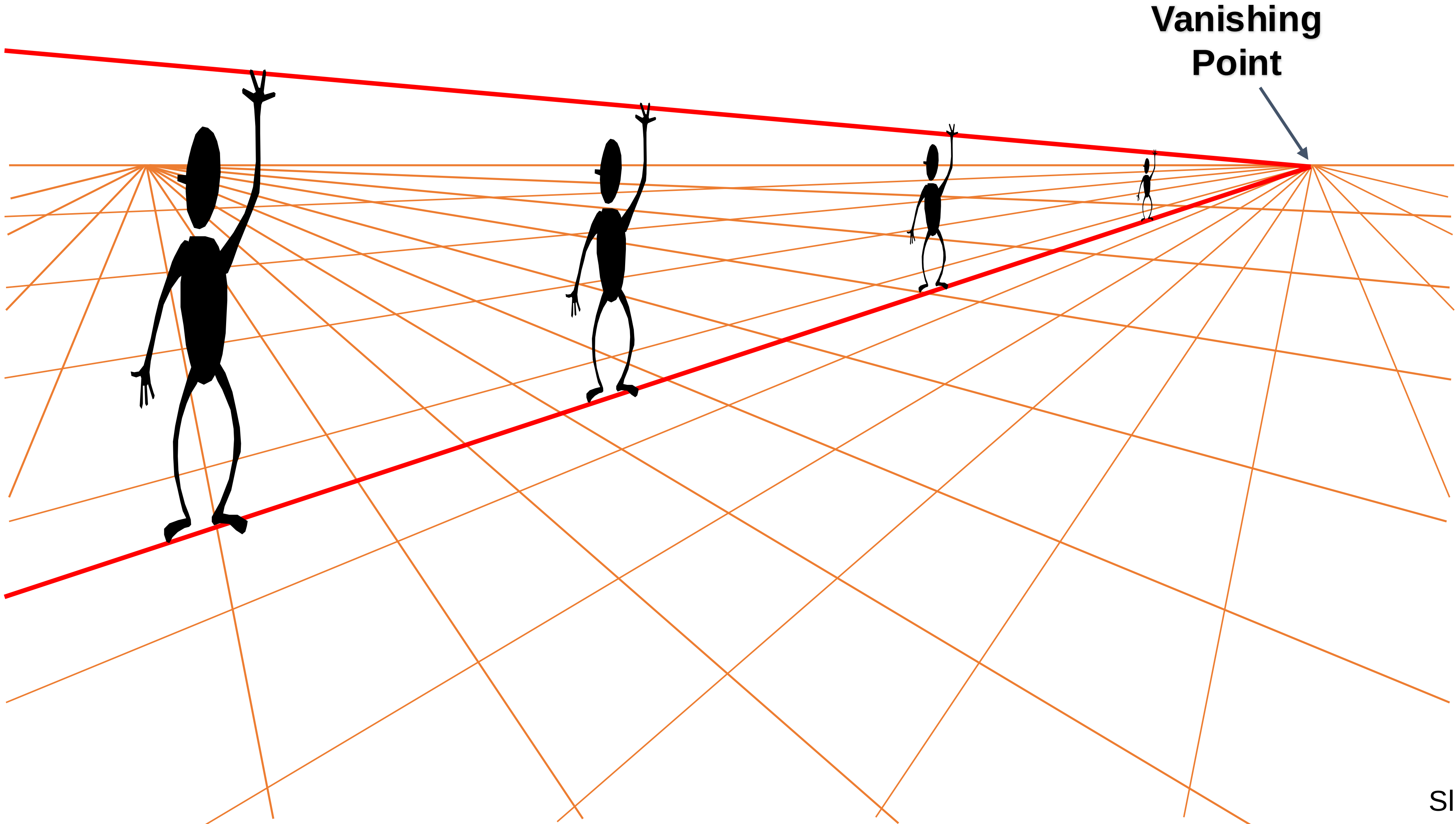
Perspective Cues



Are they of the same height?

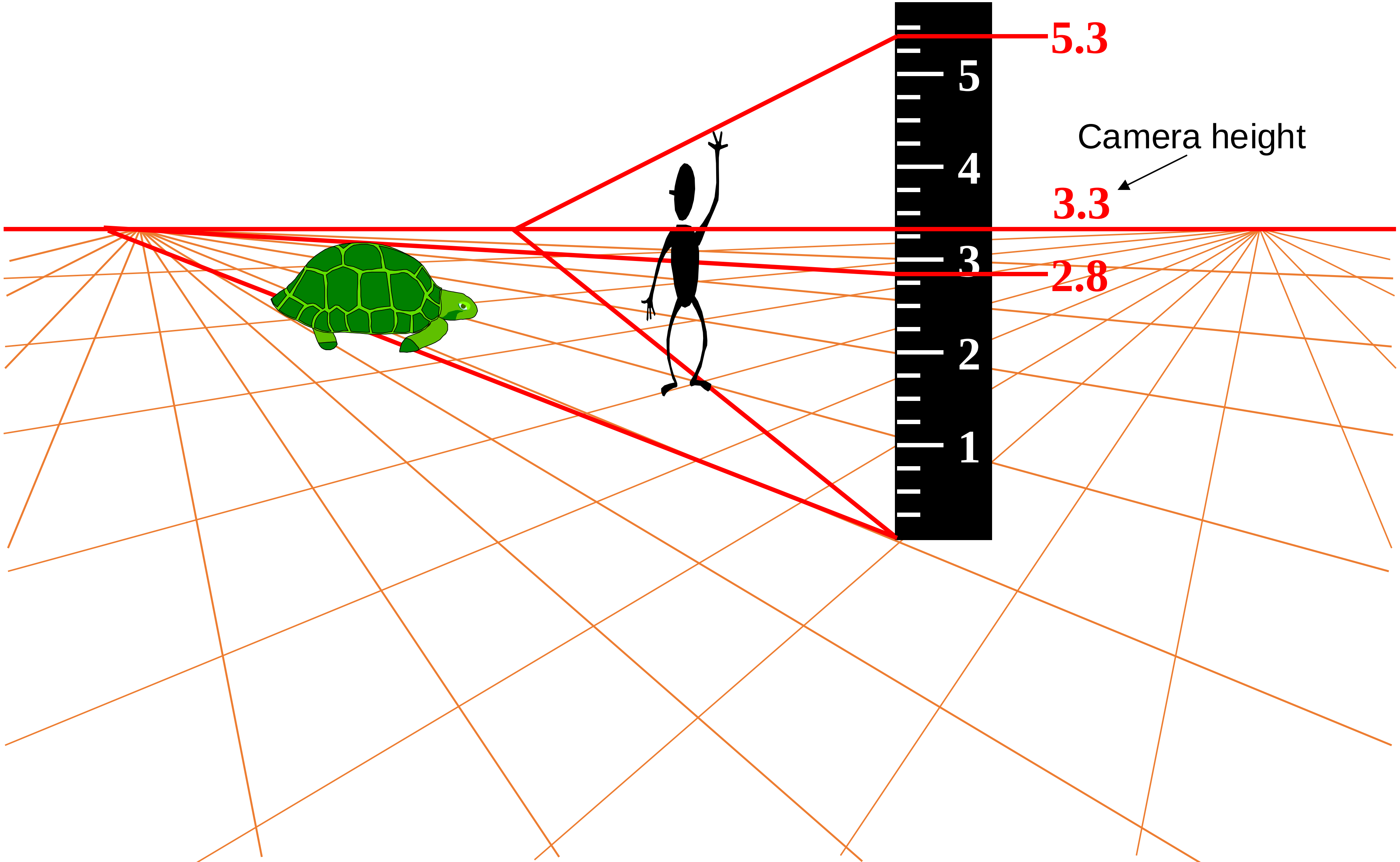
Slide by Steve Seitz

Comparing Heights

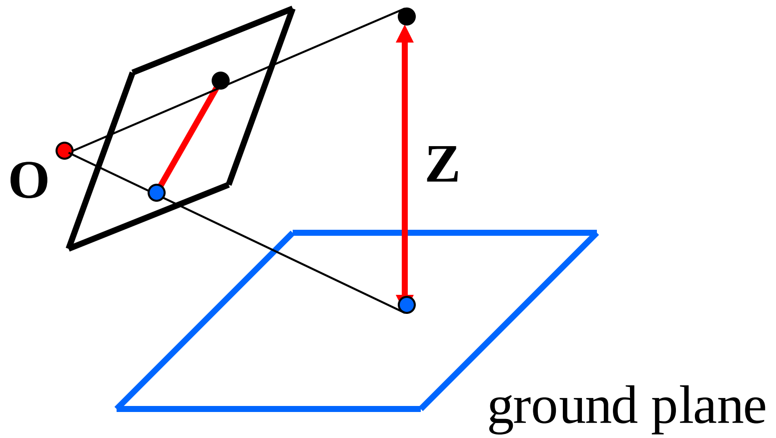


Slide by Steve Seitz

Measuring Height with a Ruler



Measuring Height without a Ruler



Compute Z from image measurements

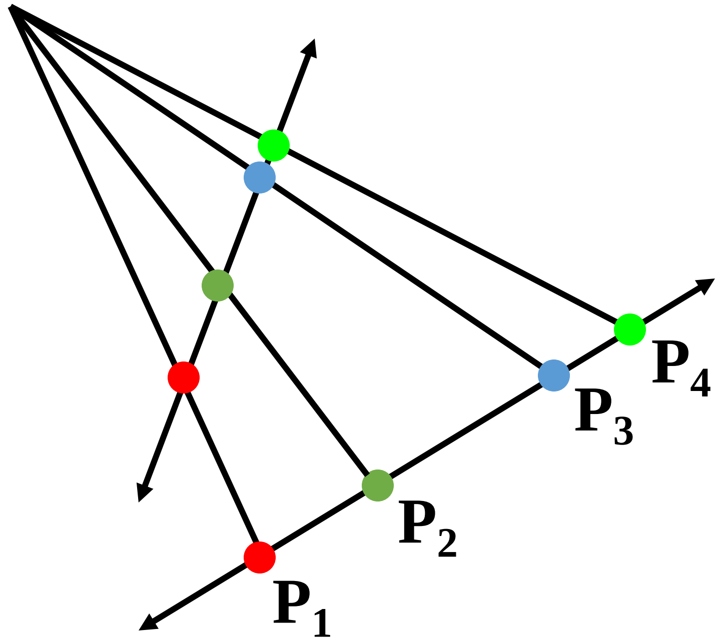
- Need more than vanishing points to do this

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?

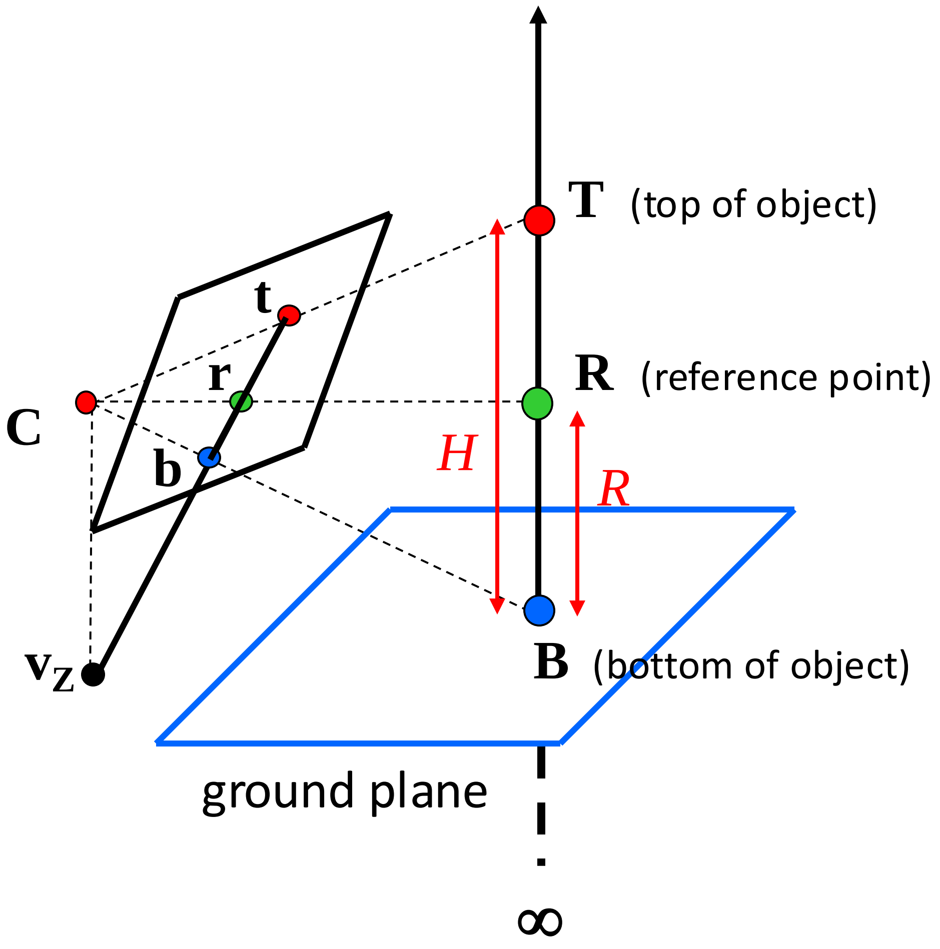
Projective Invariant

- 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:



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

Measuring Height



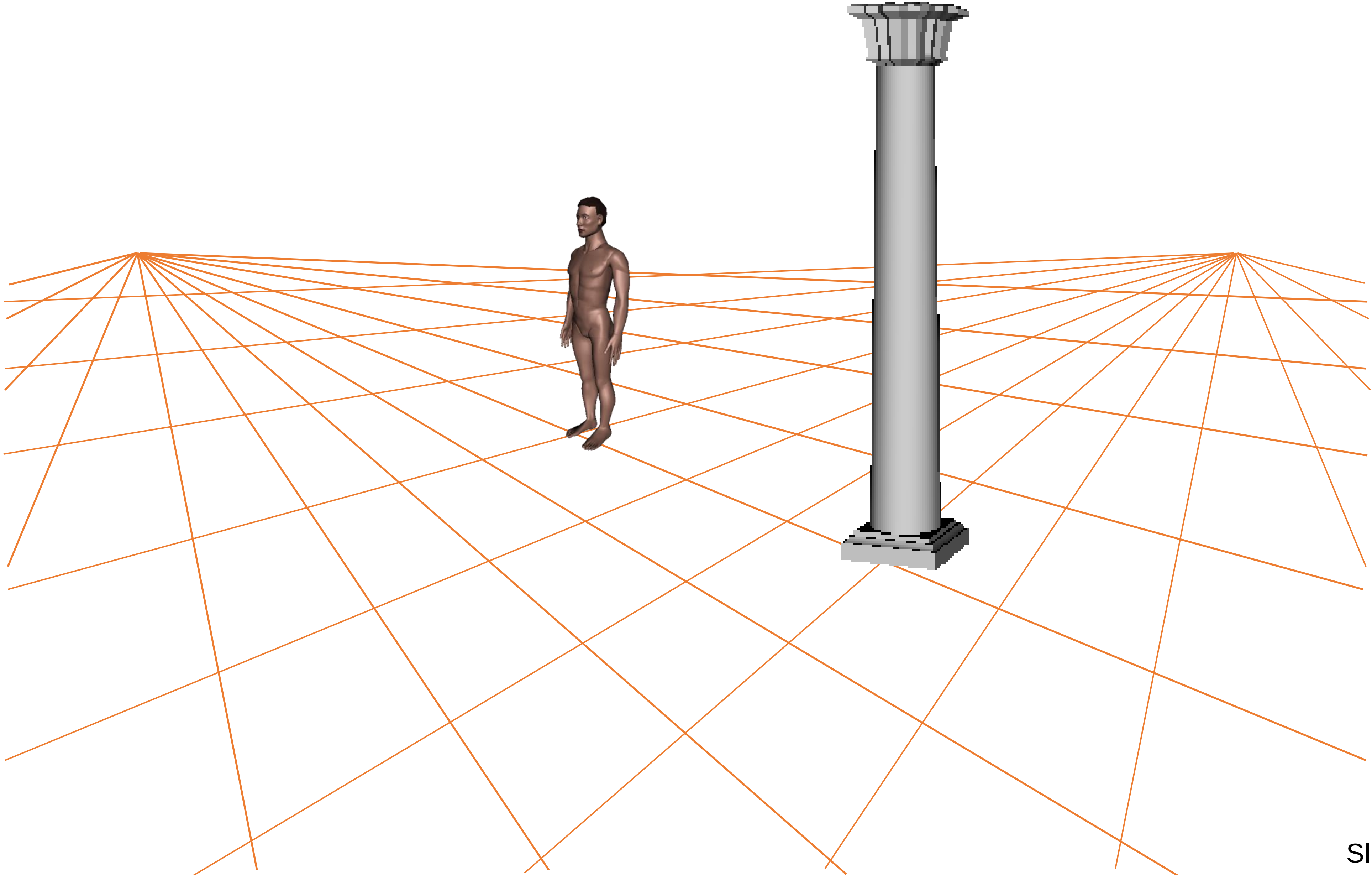
$$\frac{\|\mathbf{T}-\mathbf{B}\|\,\|\infty-\mathbf{R}\|}{\|\mathbf{R}-\mathbf{B}\|\,\|\infty-\mathbf{T}\|} = \frac{H}{R}$$

scene cross ratio

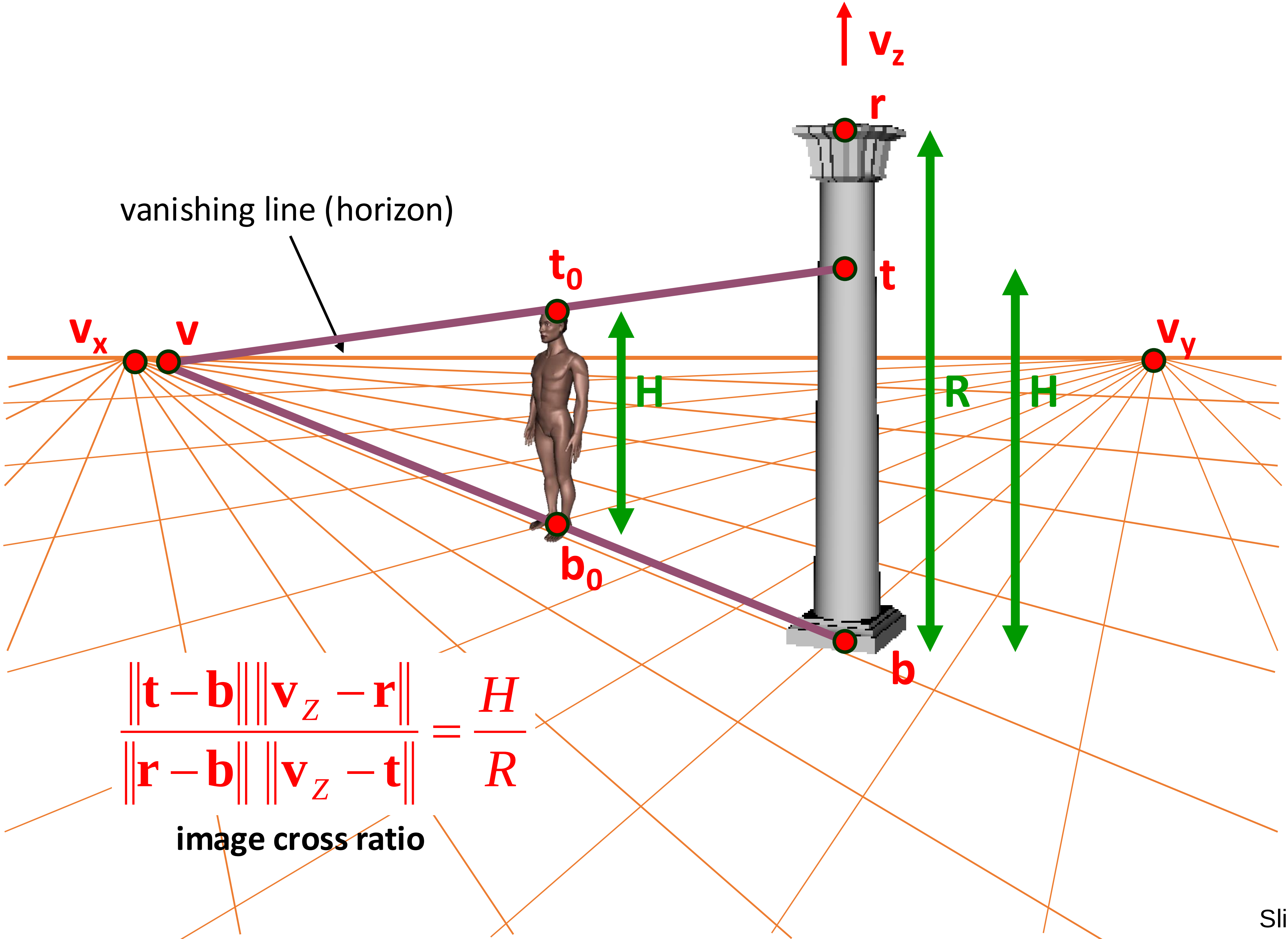
$$\frac{\|\mathbf{t}-\mathbf{b}\|\|\mathbf{v}_Z-\mathbf{r}\|}{\|\mathbf{r}-\mathbf{b}\|\,\|\mathbf{v}_Z-\mathbf{t}\|} = \frac{H}{R}$$

image cross ratio

Measuring Height without a Ruler



Slide by Steve Seitz



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

image cross ratio

Slide by Steve Seitz

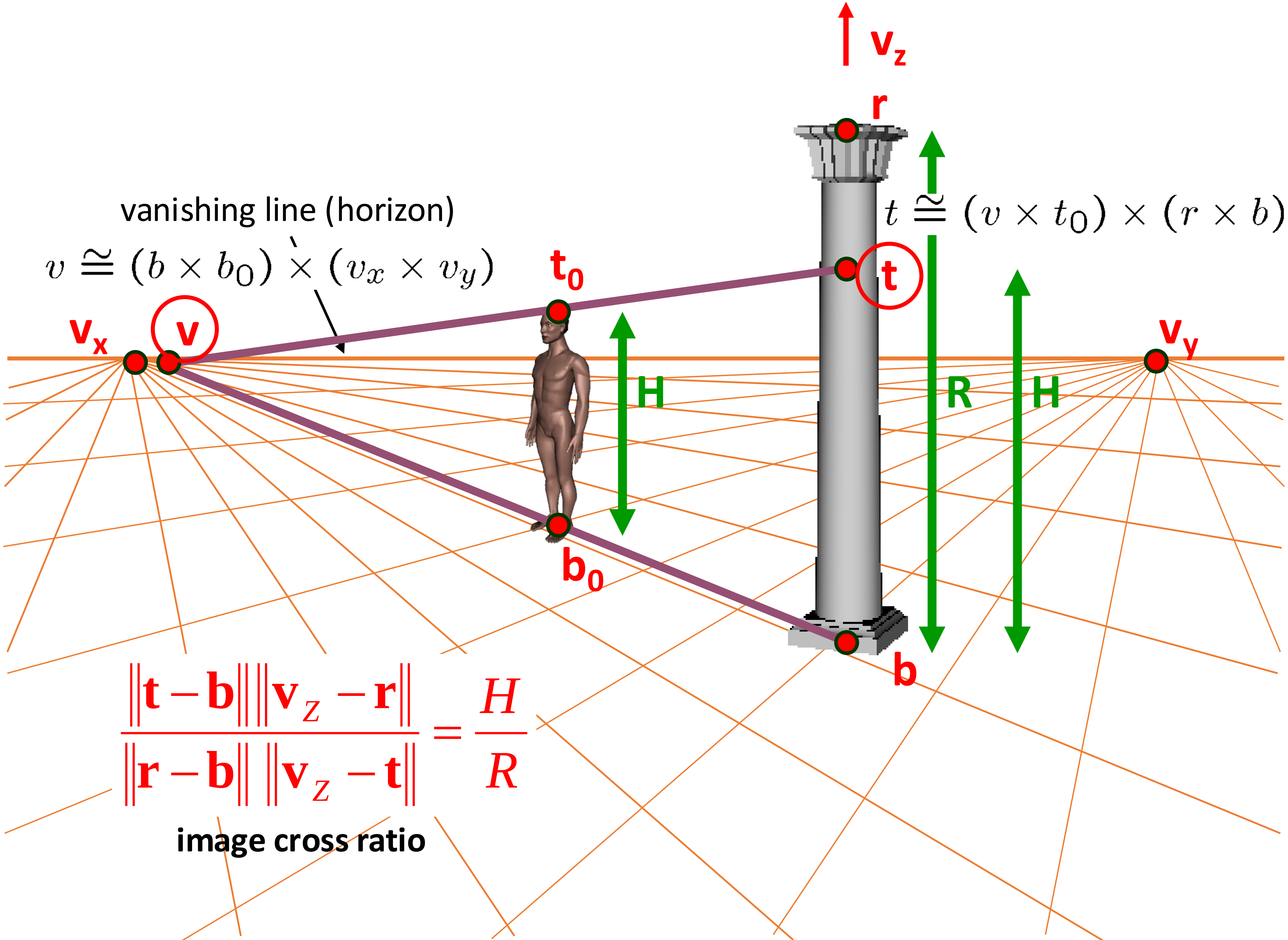
Recall: 2D lines in Homogeneous Coordinates

- 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: $\mathbf{x} = \mathbf{l}_1 \times \mathbf{l}_2$



vanishing line (horizon)

$$v \cong (b \times b_0) \times (v_x \times v_y)$$

$$t \cong (v \times t_0) \times (r \times b)$$

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

image cross ratio