CEE598 - Visual Sensing for Civil Infrastructure Eng. & Mgmt.

Session 5 – Single View Metrology

Mani Golparvar-Fard
Department of Civil and Environmental Engineering
3129D, Newmark Civil Engineering Lab
e-mail: mgolpar@illinois.edu

Reminders

- "App Turns iPhone into a Smarted Camera"
  - Technology Review:
    http://www.technologyreview.com/computing/32235/?p1=MstRcnt&a=f
    http://www.youtube.com/watch?v=b0zLgCF42Vk&feature=player_embedded
- "Google's Art Project"
  - http://www.googleartproject.com/
    (Based on an image-based reconstruction algorithm):
  - http://www.youtube.com/watch?v=RAvnJCBYHgE&feature=player_embedded

- Camera Calibration
- Wikipage
- Assignment 1 will be online Next Tuesday

Single View Metrology

Vermeer's *Music Lesson*

Reconstructions by Criminisi et al.

New Papers in this area

- Yu Chen, Duncan Robertson and Roberto Cipolla. A Practical System for Modelling Body Shapes from Single View Measurements. *Proceedings of the British Machine Vision Conference*, pages 82.1-82.11. BMVA Press, September 2011. http://dx.doi.org/10.5244/C.25.82

Outline

Single View Metrology

- Review calibration
- Points and Lines
- Vanishing Points
- Measuring Height

Reading: [HZ] Chapters 2,3,8

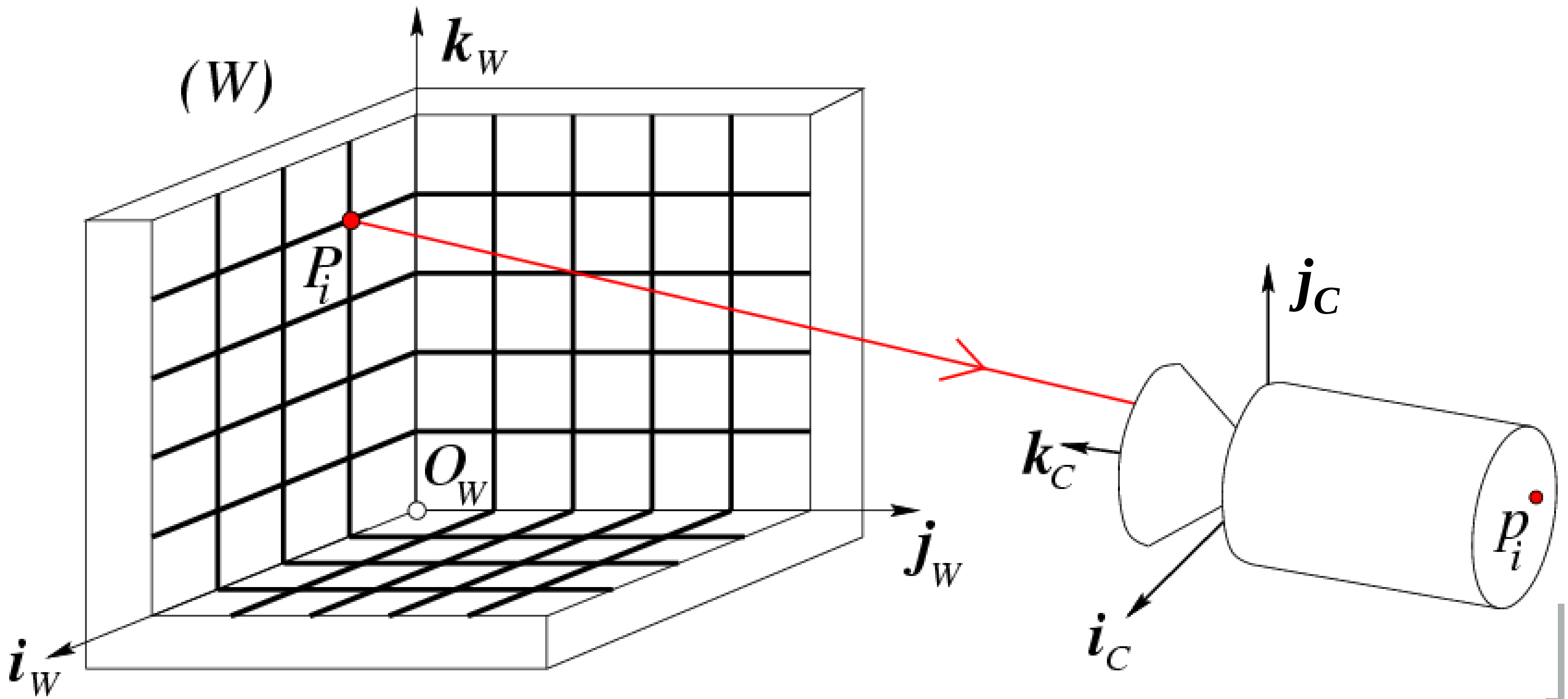
Some slides in this lecture are courtesy to Profs. D. Hoiem and S. Savarese

Outline

Single View Metrology

- Review calibration
- Points and Lines
- Vanishing Points
- Measuring Height

Calibration Problem



$$M = K\begin{bmatrix} R & T \end{bmatrix}$$

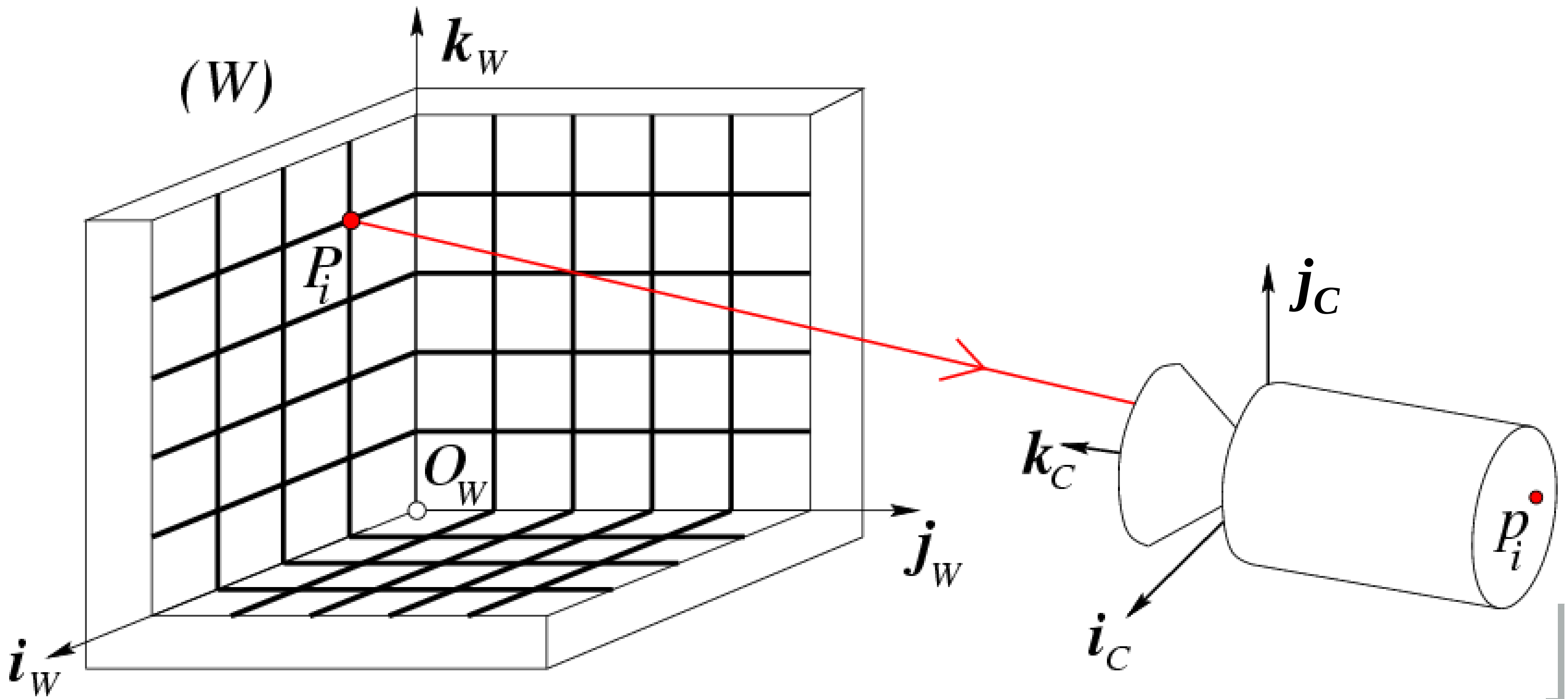
$$P_i \rightarrow M\,P_i \rightarrow p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix}$$

World ref. system

In pixels

$$K = \begin{bmatrix} \frac{1}{\lambda} & 0 & 0 \\ 0 & \frac{1}{\lambda} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha & -\alpha \cot\theta & u_o \\ 0 & \frac{\beta}{\sin\theta} & v_o \\ 0 & 0 & 1 \end{bmatrix}$$

Calibration Problem



$$M = K\begin{bmatrix} R & T \end{bmatrix}$$

$$P_i \rightarrow M\,P_i \rightarrow p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix}$$

World ref. system

In pixels

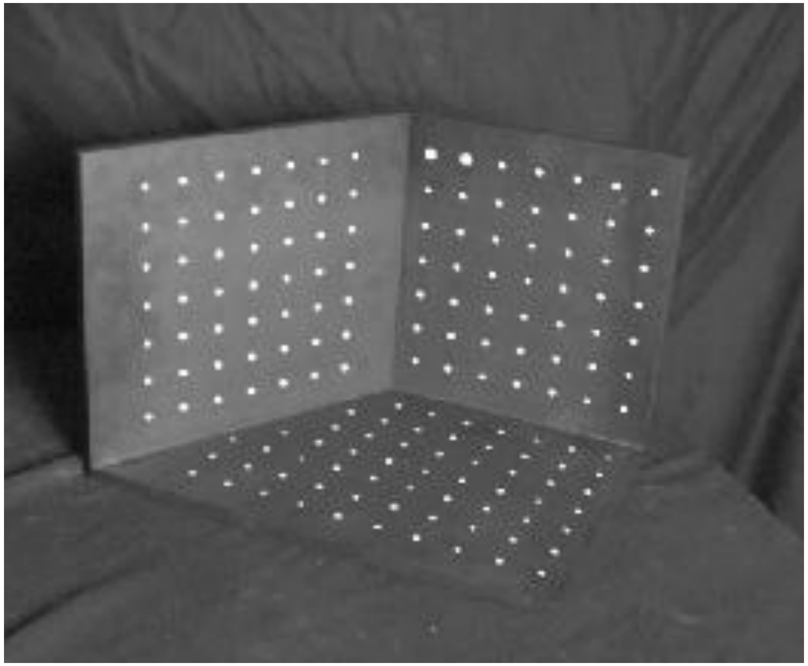
11 unknown

Need at least 6 correspondences

Calibrating the Camera

Method 1: Use an object (calibration grid) with known geometry

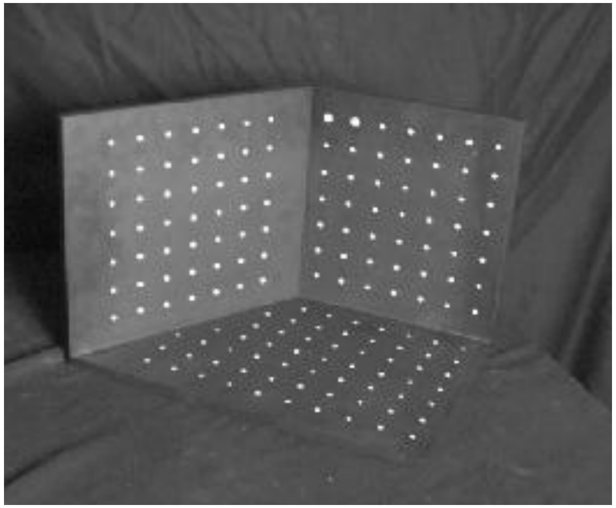
- Correspond image points to 3d points
- Get least squares solution (or non-linear solution)



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

Estimating the Projection Matrix

- Place a known object in the scene
  - Identify correspondence between image and scene
  - Compute mapping from scene to image



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

$$\mathrm{M} = \mathrm{K}\begin{bmatrix} \mathrm{R} & \mathrm{T} \end{bmatrix}$$

Direct Linear Calibration

$$\lambda \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

$$u_i = \frac{m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}}$$

$$v_i = \frac{m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}}$$

Direct Linear Calibration

$$\lambda \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

$$u_i = \frac{m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}}$$

$$v_i = \frac{m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}}$$

$$u_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}) = m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}$$
$$v_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}) = m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}$$

Direct Linear Calibration

$$\lambda \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

$$u_i = \frac{m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}}$$

$$v_i = \frac{m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}}$$

$$u_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}) = m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}$$
$$v_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}) = m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}$$

$$\begin{bmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -u_iX_i & -u_iY_i & -u_iZ_i & -u_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -v_iX_i & -v_iY_i & -v_iZ_i & -v_i \end{bmatrix} \begin{bmatrix} m_{00} \\ m_{01} \\ m_{02} \\ m_{03} \\ m_{10} \\ m_{11} \\ m_{12} \\ m_{13} \\ m_{20} \\ m_{21} \\ m_{22} \\ m_{23} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Ax=0 form

Direct Linear Calibration

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 & -u_1Z_1 & -u_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1X_1 & -v_1Y_1 & -v_1Z_1 & -v_1 \\ & & & & & & \vdots & & & & & \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & -u_nX_n & -u_nY_n & -u_nZ_n & -u_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -v_nX_n & -v_nY_n & -v_nZ_n & -v_n \end{bmatrix} \begin{bmatrix} m_{00} \\ m_{01} \\ m_{02} \\ m_{03} \\ m_{10} \\ m_{11} \\ m_{12} \\ m_{13} \\ m_{20} \\ m_{21} \\ m_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

Ax=0 form

Can solve for m_{ij} by linear least squares

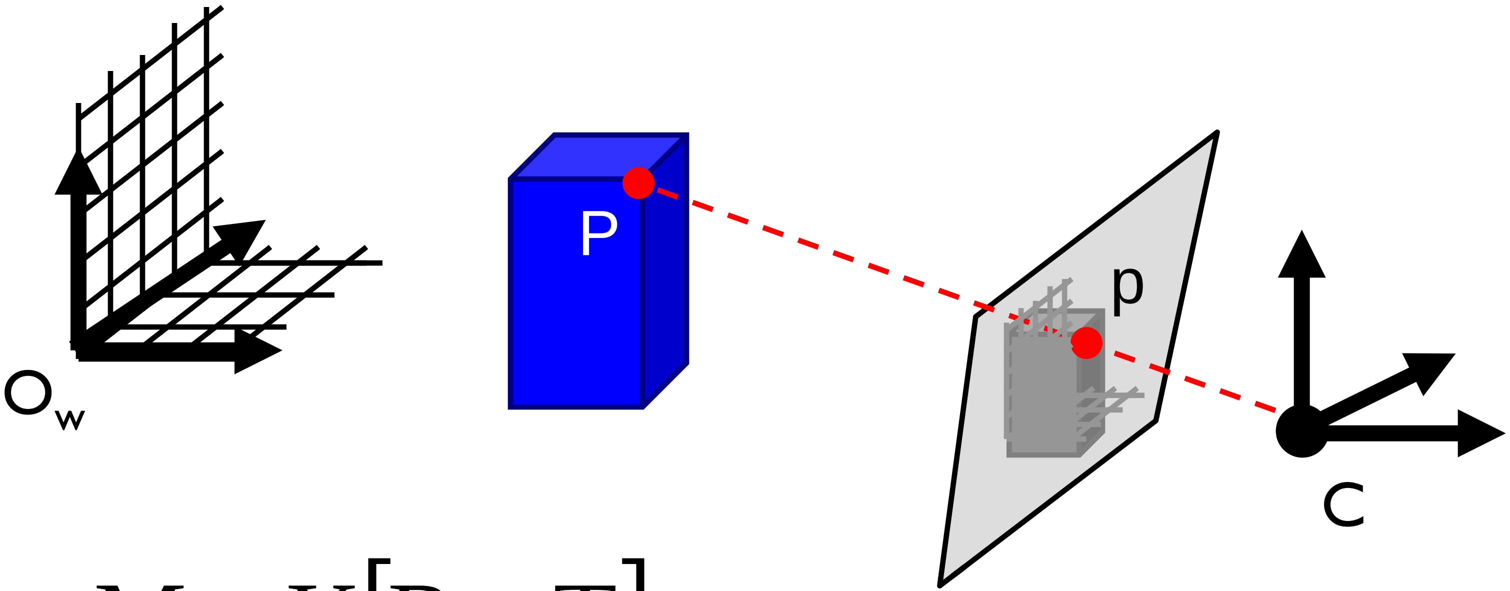
Calibration with linear method

- Advantages: easy to formulate and solve

- Disadvantages
  - Doesn't tell you camera parameters
  - Doesn't model radial distortion
  - Can't impose constraints, such as known focal length
  - Doesn't minimize right error function (see HZ p. 181)

- Non-linear methods are preferred
  - Define error as difference between projected points and measured points
  - Minimize error using Newton's method or other non-linear optimization

Once the camera is calibrated...



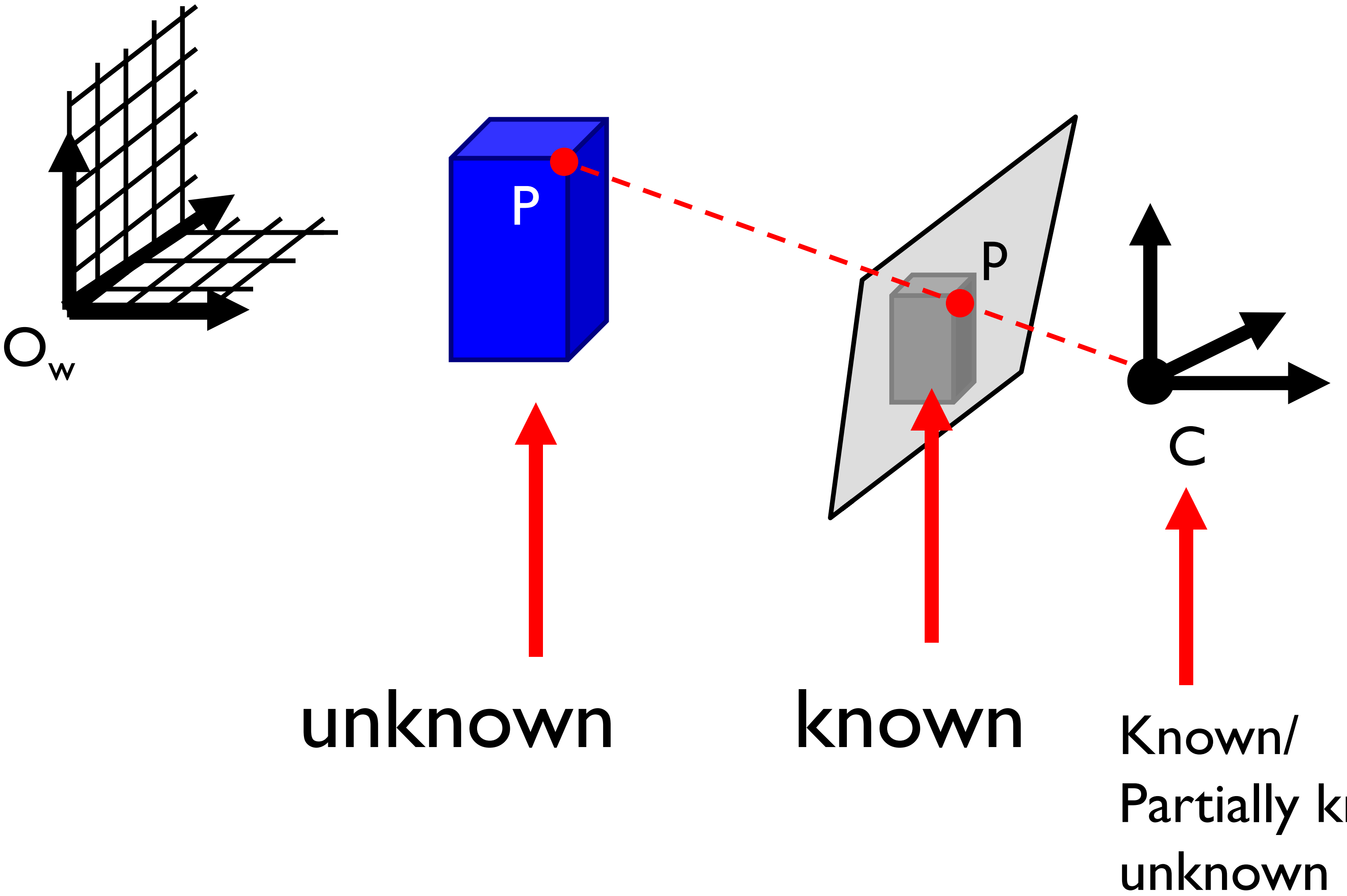
$$M = K\begin{bmatrix} R & T \end{bmatrix}$$

-Internal parameters K are known
-R,T are known – but these can only relate C to the calibration rig

Can I estimate P from the measurement p from a single image?

No - in general ☹ [P can be anywhere along the line defined by C and p]

Recovering structure from a single view



Outline

Single View Metrology

- Review calibration
- Points and Lines
- Vanishing Points
- Measuring Height

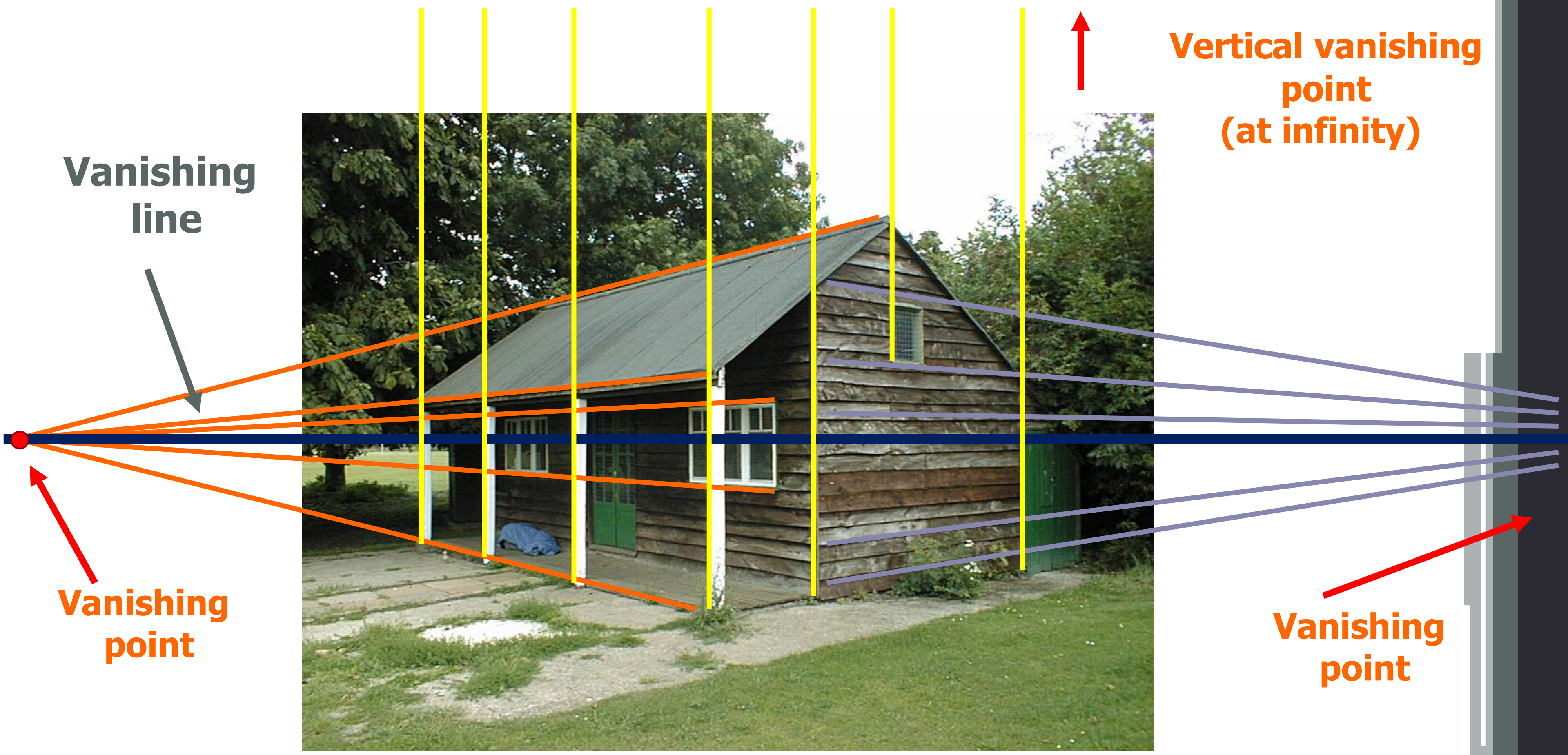
Camera Calibration

- What if world coordinates are not known?
- Can we use scene features(vanishing points)?

Calibrating the Camera

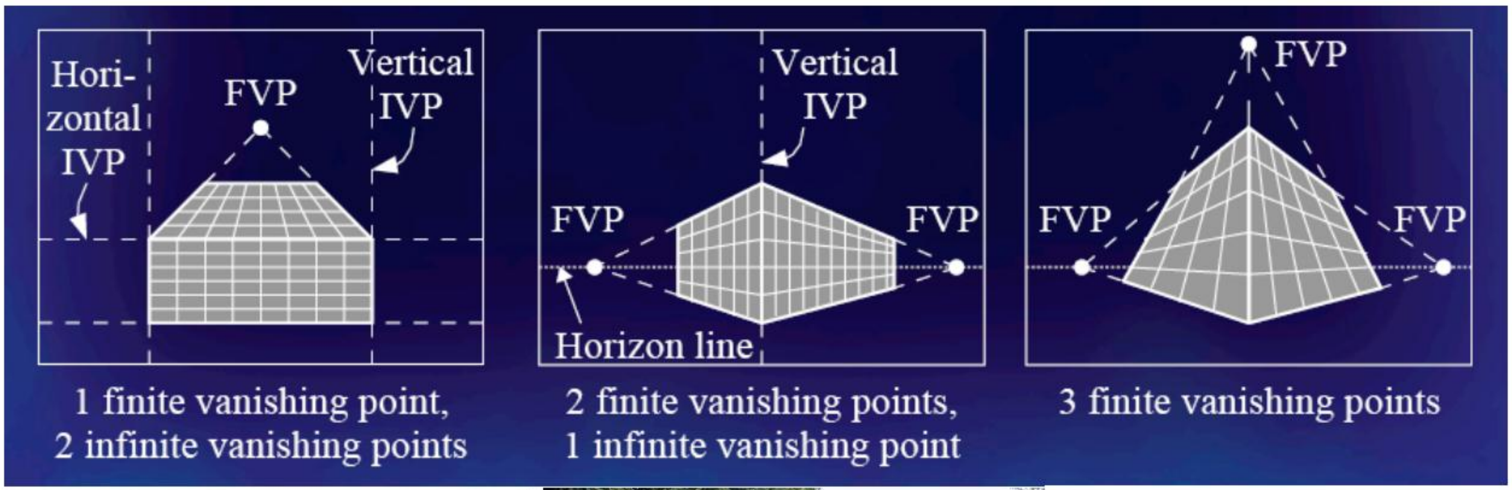
Method 2: Use vanishing points

- Find vanishing points corresponding to orthogonal directions



Vanishing Points and Lines

- Scene contains lines along directions that are orthogonal



Calibration by orthogonal vanishing points

- Intrinsic camera matrix
  - Use orthogonality as a constraint
  - Model K with only f, u_0, v_0

$$\mathbf{p}_i = \mathbf{KRX}_i$$

For vanishing points

$$\mathbf{X}_i^T \mathbf{X}_j = 0$$

- What if you don't have three finite vanishing points?
  - Two finite VP: solve f, get valid u_0, v_0 closest to image center
  - One finite VP: u_0, v_0 is at vanishing point; can't solve for f

Calibration by vanishing points

- Intrinsic camera matrix

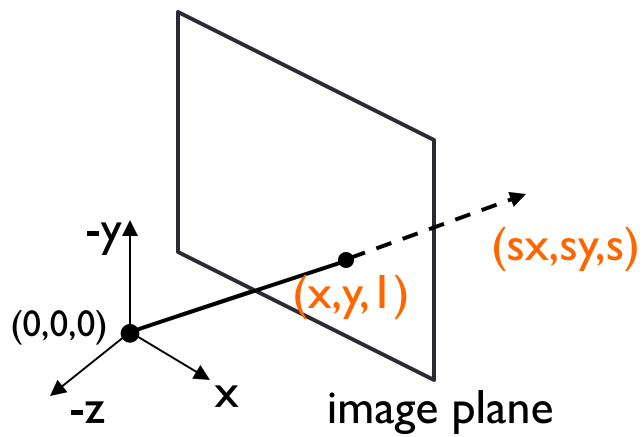
$$\mathbf{p}_i = \mathbf{KRX}_i$$

- Rotation matrix
  - Set directions of vanishing points
    - e.g., $\mathbf{X}_1 = [1, 0, 0]$
  - Each VP provides one column of **R**
  - Special properties of **R**
    - inv(**R**)=$\mathbf{R}^T$
    - Each row and column of **R** has unit length

The projective plane

- Why do we need homogeneous coordinates?
  - represent points at infinity, homographies, perspective projection, multi-view relationships

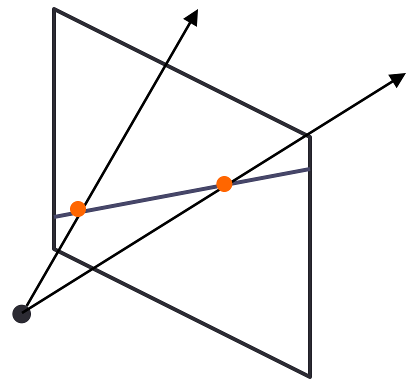
- What is the geometric intuition?
  - a point in the image is a *ray* in projective space



- Each *point* **(x,y)** on the plane is represented by a *ray* **(sx,sy,s)**
  - all points on the ray are equivalent: **(x, y, 1) ≅ (sx, sy, s)**

Projective lines

- What does a line in the image correspond to in projective space?



- A line is a *plane* of rays through origin
  - all rays (x,y,z) satisfying: ax + by + cz = 0

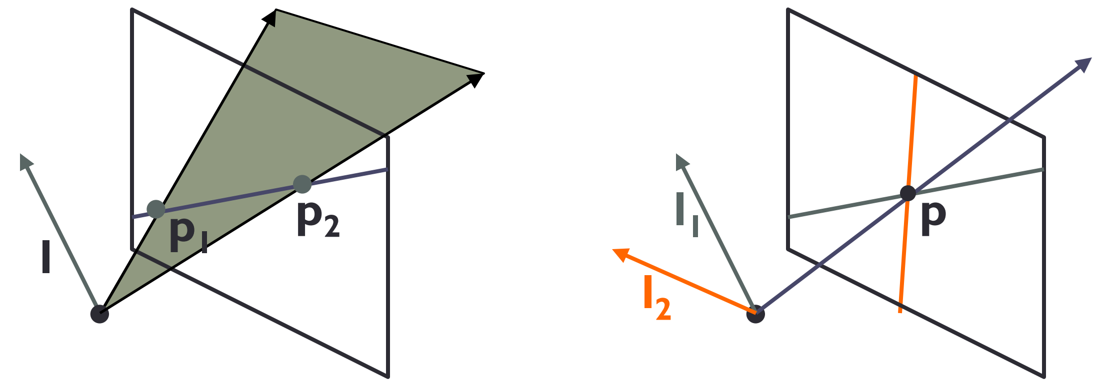
$$\text{in vector notation:} \quad 0 = \begin{bmatrix} a & b & c \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

l **p**

- A line is also represented as a homogeneous 3-vector **l**

Point and line duality

- A line **l** is a homogeneous 3-vector
- It is $\perp$ to every point (ray) **p** on the line: **l p**=0



What is the line **l** spanned by rays $\mathbf{p}_1$ and $\mathbf{p}_2$?

- **l** is $\perp$ to $\mathbf{p}_1$ and $\mathbf{p}_2$ $\Rightarrow$ $\mathbf{l} = \mathbf{p}_1 \times \mathbf{p}_2$
- **l** is the plane normal

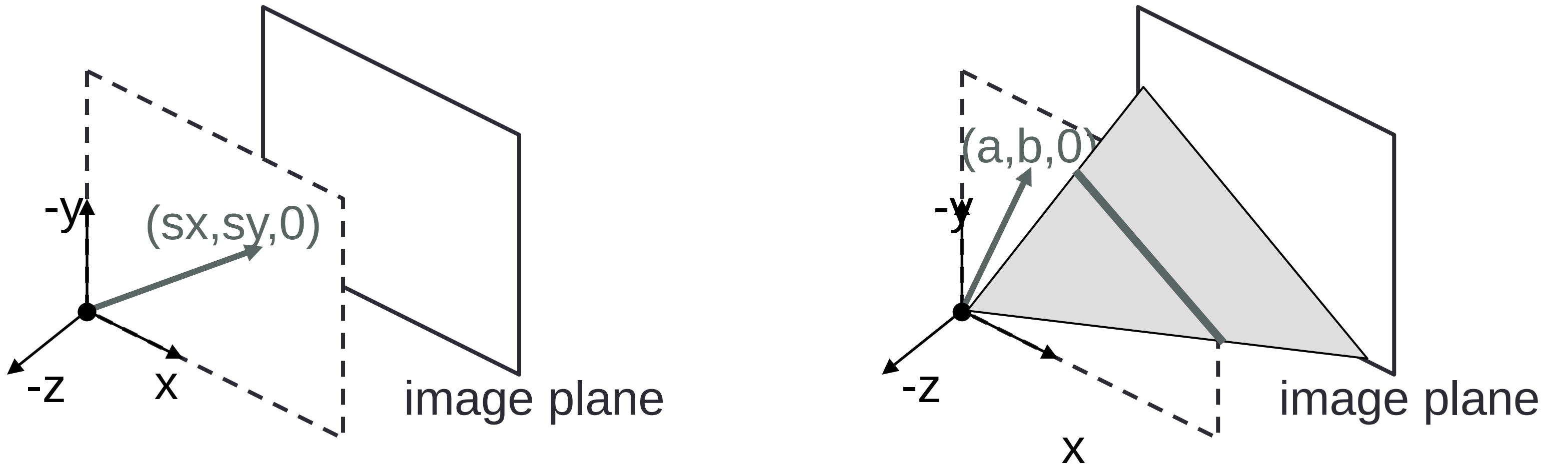
What is the intersection of two lines $\mathbf{l}_1$ and $\mathbf{l}_2$?

- **p** is $\perp$ to $\mathbf{l}_1$ and $\mathbf{l}_2$ $\Rightarrow$ $\mathbf{p} = \mathbf{l}_1 \times \mathbf{l}_2$

Points and lines are *dual* in projective space

- given any formula, can switch the meanings of points and lines to get another formula

Ideal points and lines



- Ideal point ("point at infinity")
  - p ≅ (x, y, 0) – parallel to image plane
  - It has infinite image coordinates

Ideal line

- l ≅ (a, b, 0) – parallel to image plane
- Corresponds to a line in the image (finite coordinates)
  - goes through image origin (*principle point*)

Homographies of points and lines

- Computed by 3x3 matrix multiplication
  - To transform a point: $\boldsymbol{p}' = \boldsymbol{Hp}$
  - To transform a line: $\boldsymbol{lp} = 0 \to l'p' = 0$

$$0 = \boldsymbol{lp} = \boldsymbol{lH^{-1}Hp} = \boldsymbol{lH^{-1}p'} \Rightarrow \boldsymbol{l'} = \boldsymbol{lH^{-1}}$$

lines are transformed by post multiplication of $\boldsymbol{H^{-1}}$

3D projective geometry

These concepts generalize naturally to 3D

- Homogeneous coordinates
  - Projective 3D points have four coords: $\boldsymbol{P} = (X, Y, Z, W)$
- Duality
  - A plane $\boldsymbol{N}$ is also represented by a 4-vector
  - Points and planes are dual in 3D: $\boldsymbol{N}\,\boldsymbol{P} = 0$

- Projective transformations
  - Represented by 4x4 matrices $\boldsymbol{T}$: $\boldsymbol{P}' = \boldsymbol{TP}, \quad \boldsymbol{N}' = \boldsymbol{N}\,T^{-1}$

3D to 2D: "perspective" projection

- Matrix Projection:

$$\mathbf{p} = \begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{\Pi P}$$

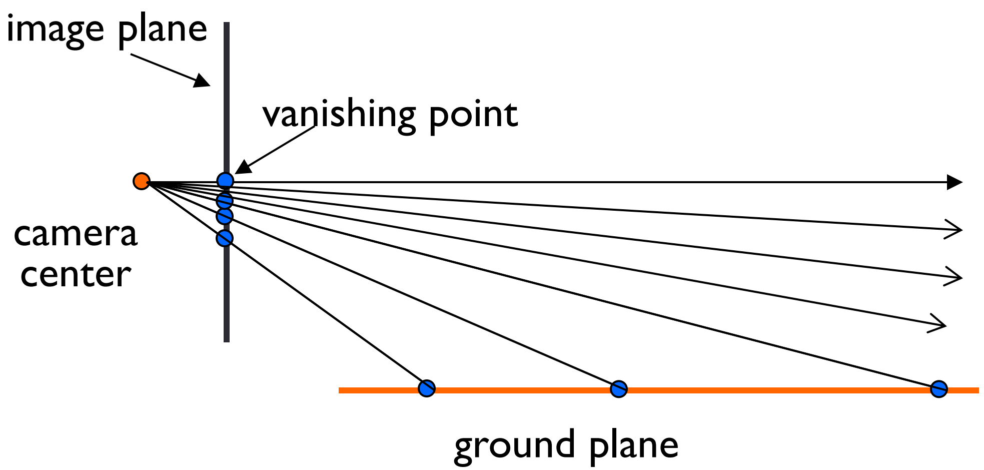
What is *not* preserved under perspective projection?

- Length, angles, parallelism

What IS preserved?

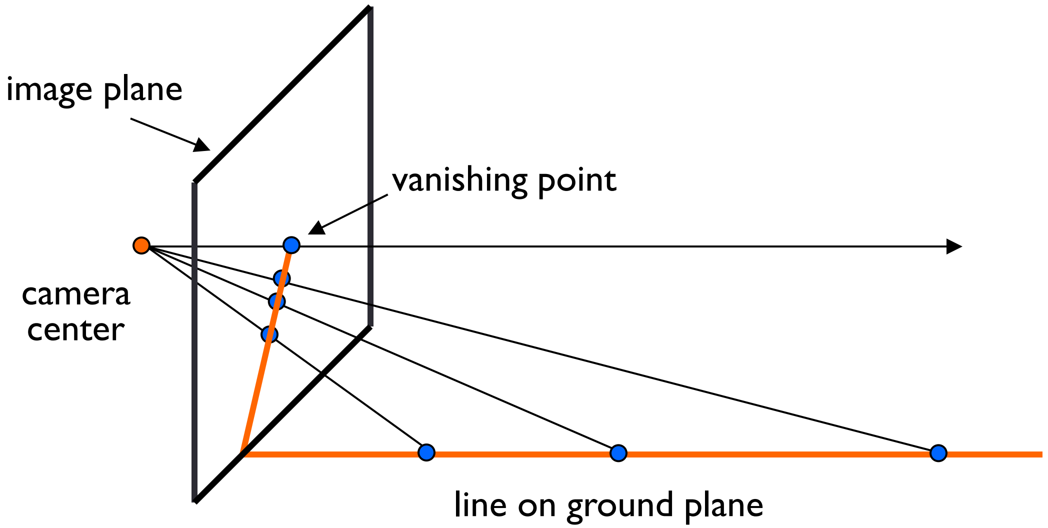
- Lines, incidences

Vanishing points

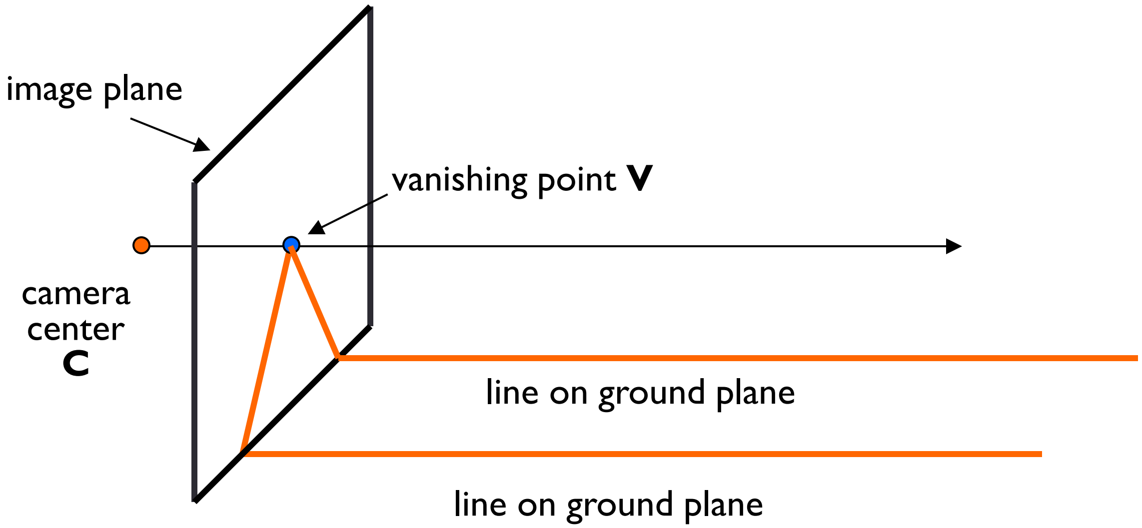


- Vanishing point
  - projection of a point at infinity

Vanishing points (2D)

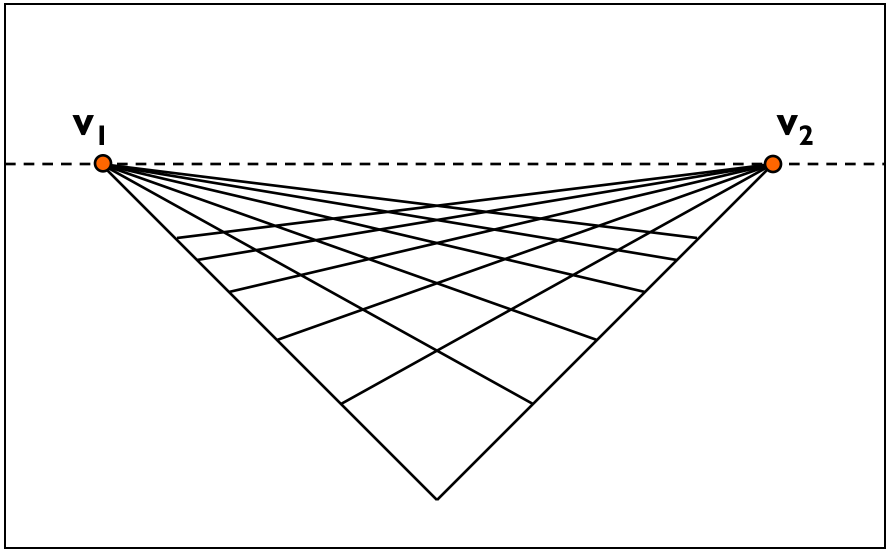


Vanishing points



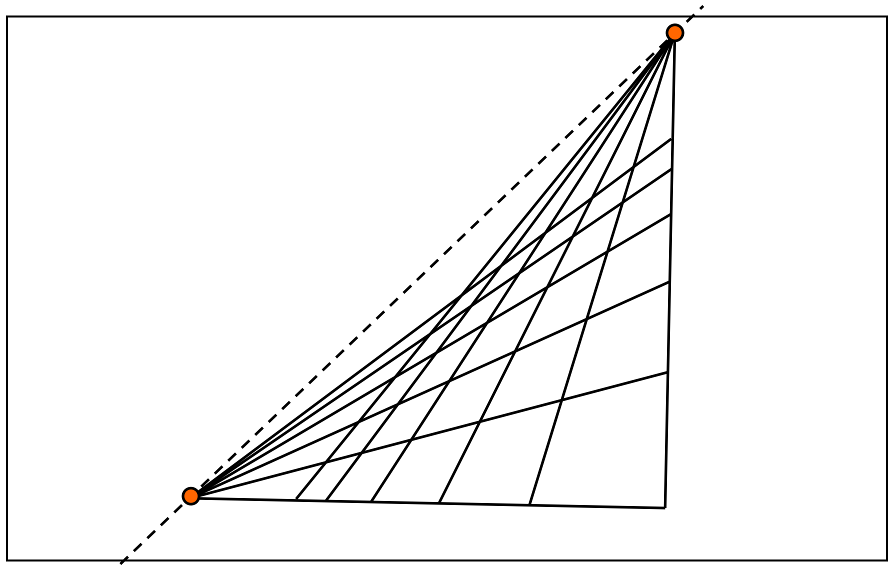
- Properties
  - Any two parallel lines have the same vanishing point **v**
  - The ray from **C** through **v** is parallel to the lines
  - An image may have more than one vanishing point
    - in fact every pixel is a potential vanishing point

Vanishing lines



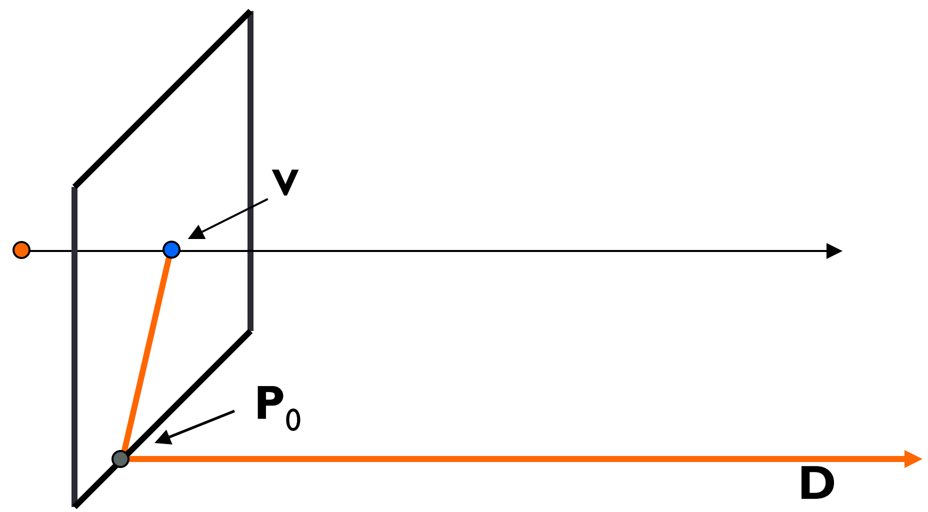
- Multiple Vanishing Points
  - Any set of parallel lines on the plane define a vanishing point
  - The union of all of these vanishing points is the *horizon line*
    - also called *vanishing line*
  - Note that different planes define different vanishing lines

Vanishing lines



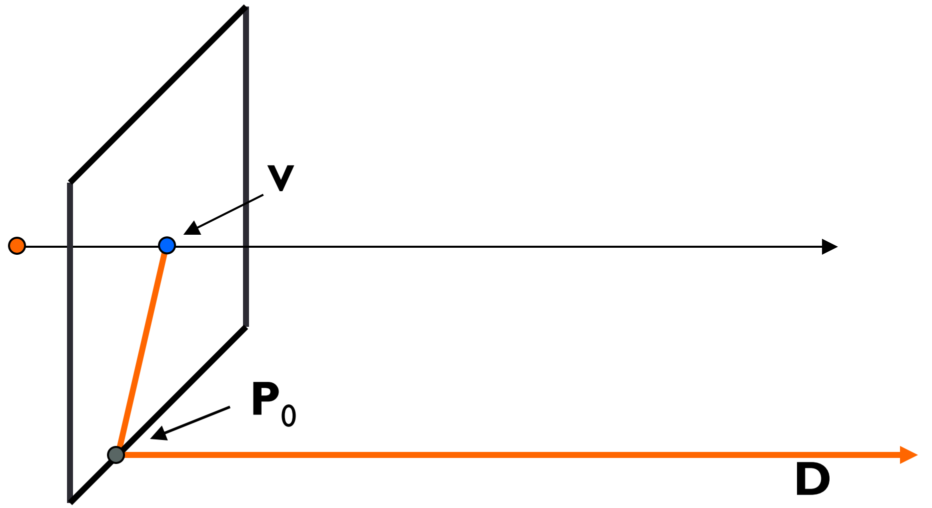
- Multiple Vanishing Points
  - Any set of parallel lines on the plane define a vanishing point
  - The union of all of these vanishing points is the *horizon line*
    - also called *vanishing line*
  - Note that different planes define different vanishing lines

Computing vanishing points



$$\mathbf{P} = \mathbf{P}_0 + t\mathbf{D}$$

Computing vanishing points

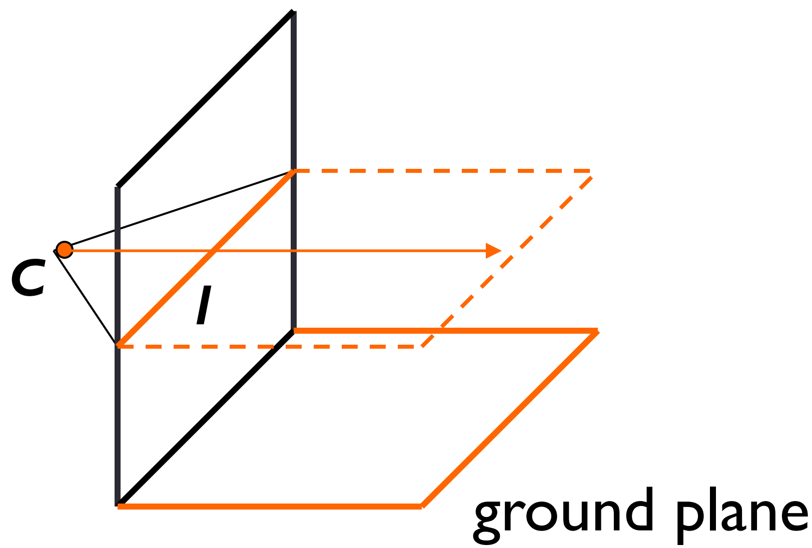


$$\mathbf{P} = \mathbf{P}_0 + t\mathbf{D}$$

$$\mathbf{P}_t = \begin{bmatrix} P_X + tD_X \\ P_Y + tD_Y \\ P_Z + tD_Z \\ 1 \end{bmatrix} \cong \begin{bmatrix} P_X / t + D_X \\ P_Y / t + D_Y \\ P_Z / t + D_Z \\ 1/t \end{bmatrix} \qquad t \to \infty \qquad \mathbf{P}_\infty \cong \begin{bmatrix} D_X \\ D_Y \\ D_Z \\ 0 \end{bmatrix}$$

- Properties $\mathbf{v} = \mathbf{\Pi P}_\infty$
  - $\boldsymbol{P}_\infty$ is a point at *infinity*, $\boldsymbol{v}$ is its projection
  - They depend only on line *direction*
  - Parallel lines $\boldsymbol{P}_0 + t\boldsymbol{D}, \boldsymbol{P}_1 + t\boldsymbol{D}$ intersect at $\boldsymbol{P}_\infty$

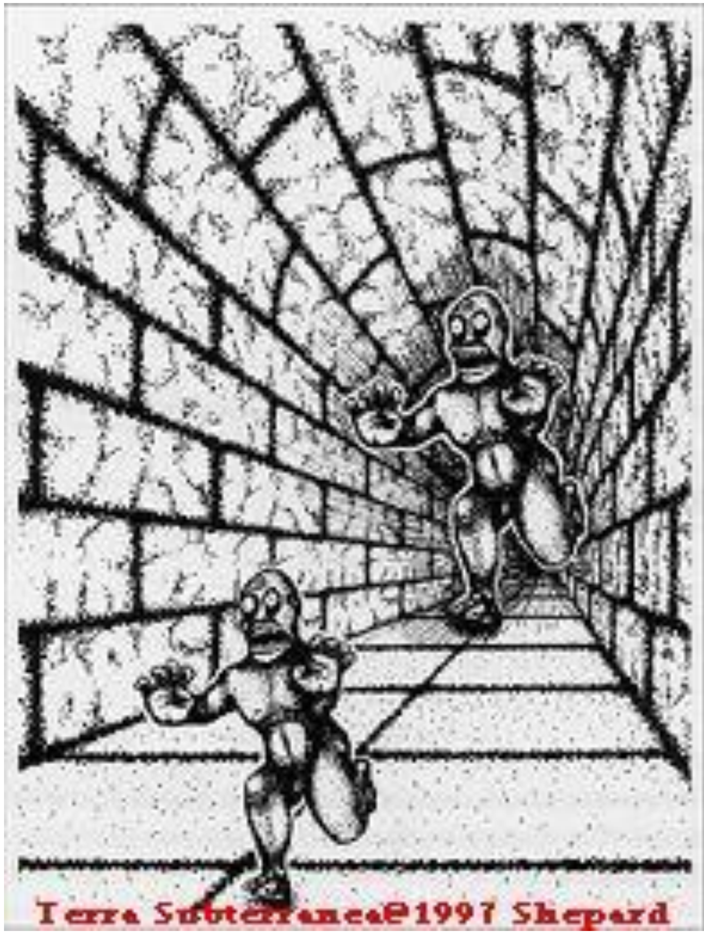
Computing vanishing lines

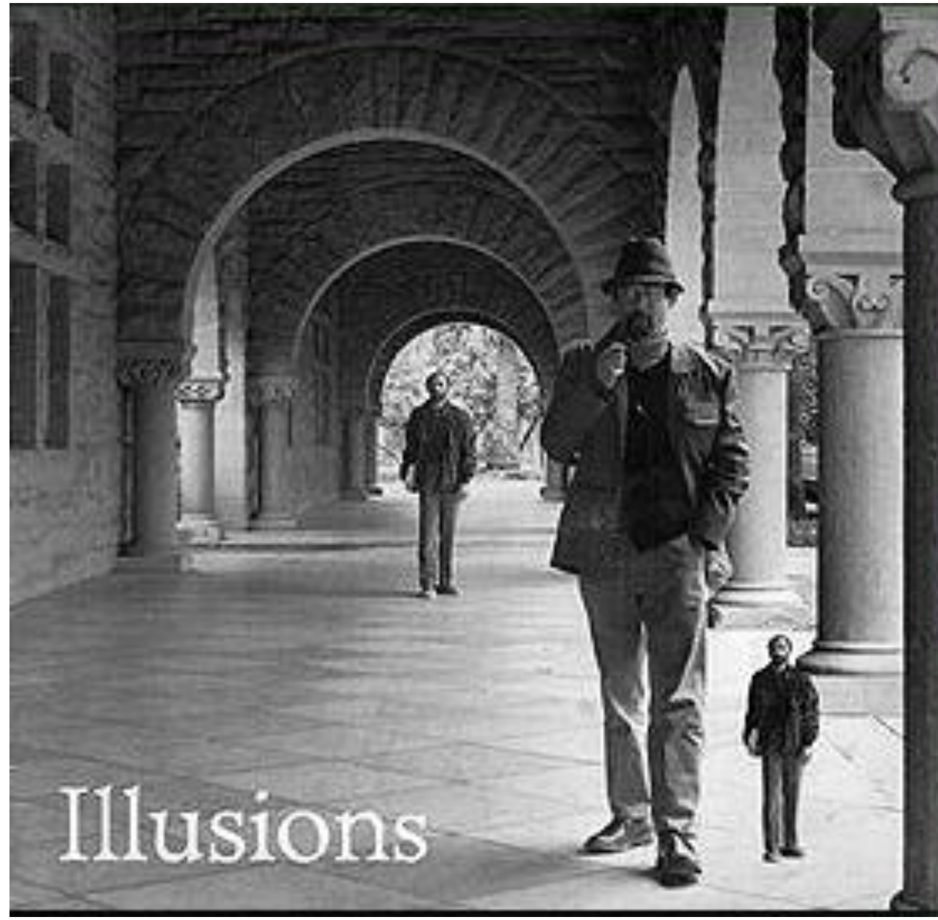


- Properties
  - $\boldsymbol{l}$ is intersection of horizontal plane through $\boldsymbol{C}$ with image plane
  - Compute $\boldsymbol{l}$ from two sets of parallel lines on ground plane
  - All points at same height as $\boldsymbol{C}$ project to $\boldsymbol{l}$
    - points higher than C project above l
  - Provides way of comparing height of objects in the scene

Is the Parachute higher than the person who is taking this picture?

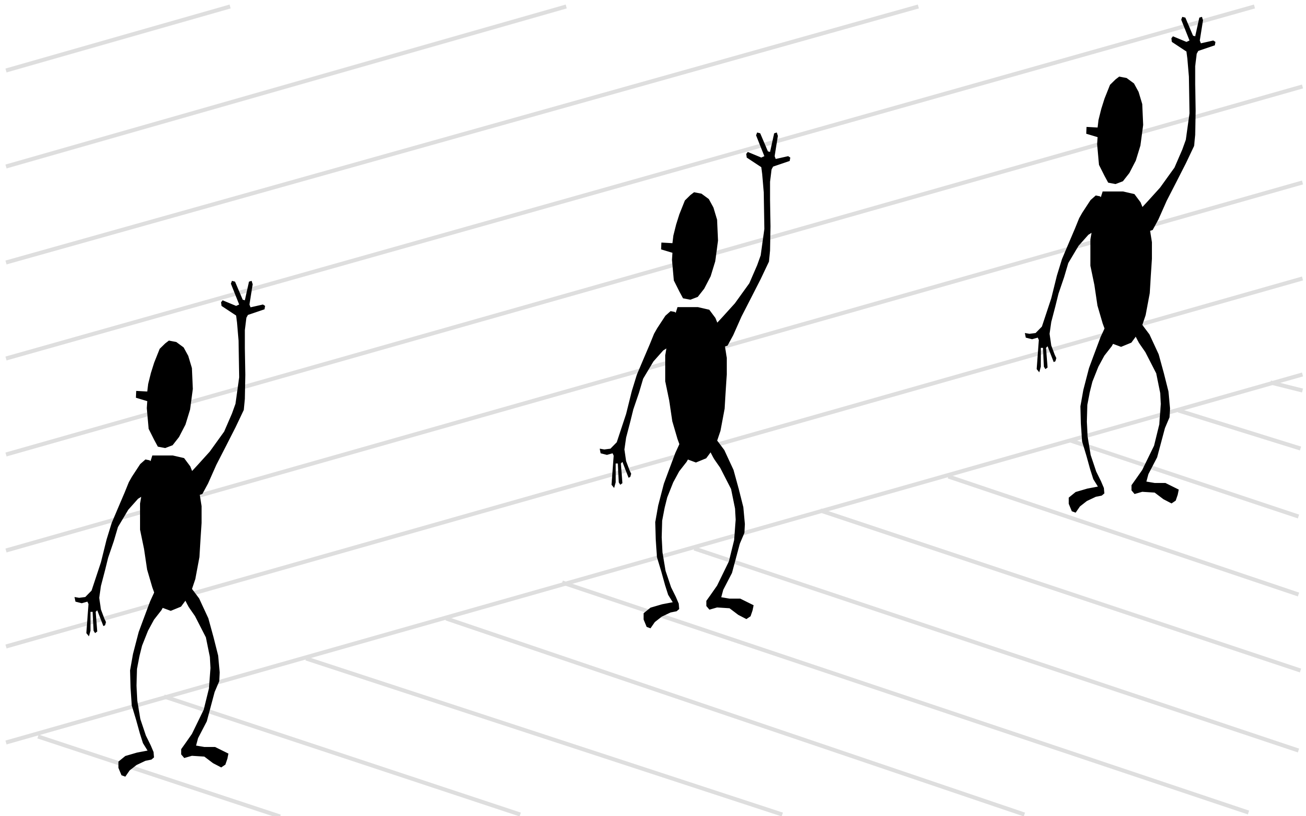
Fun with vanishing points



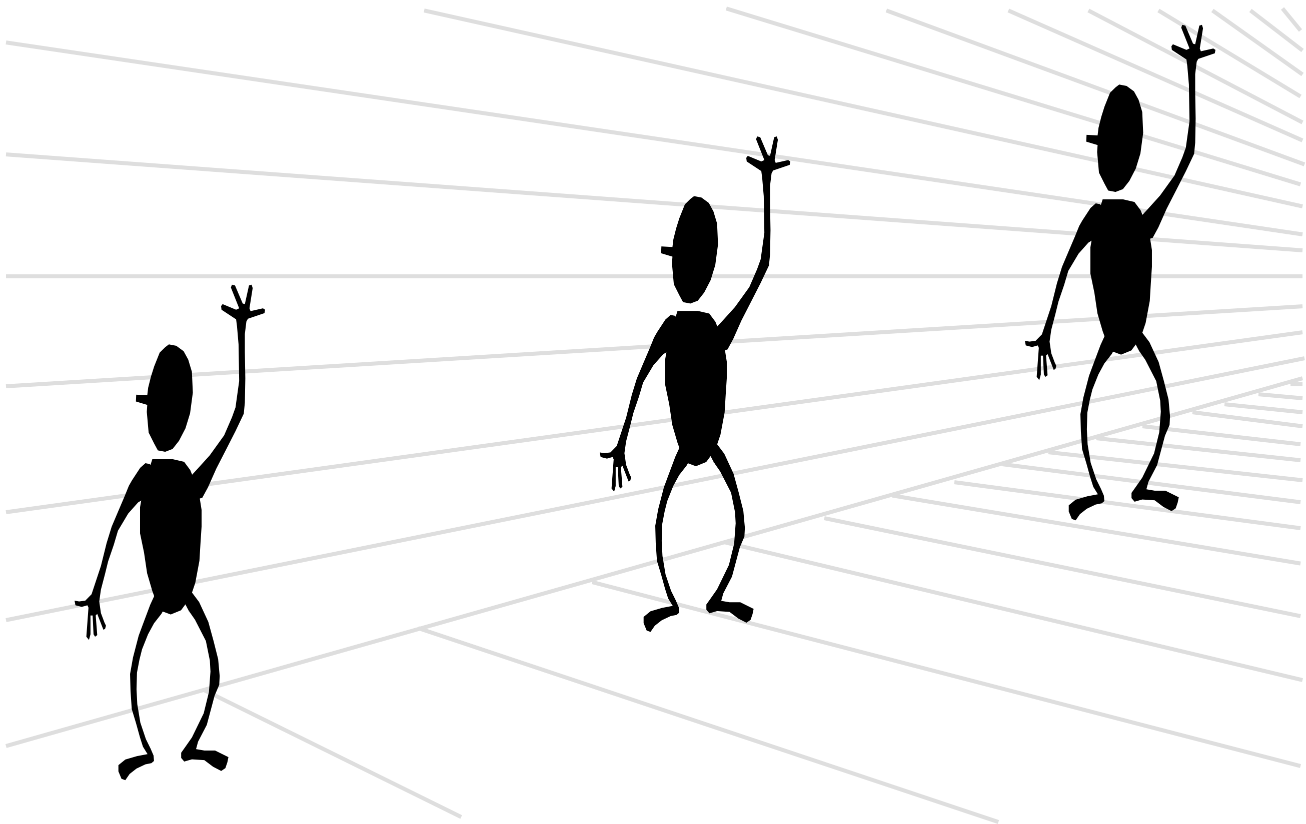


- Perspective Cues?

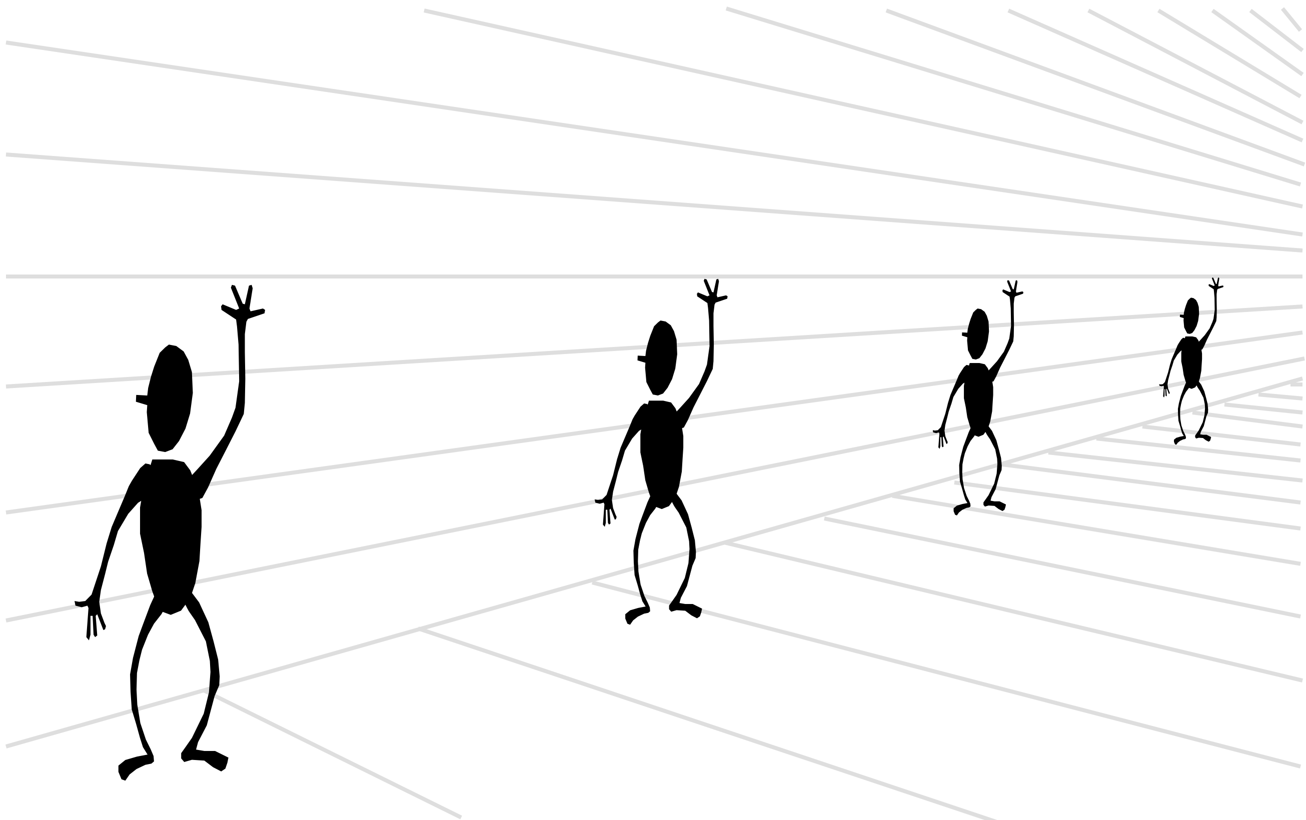
Perspective cues



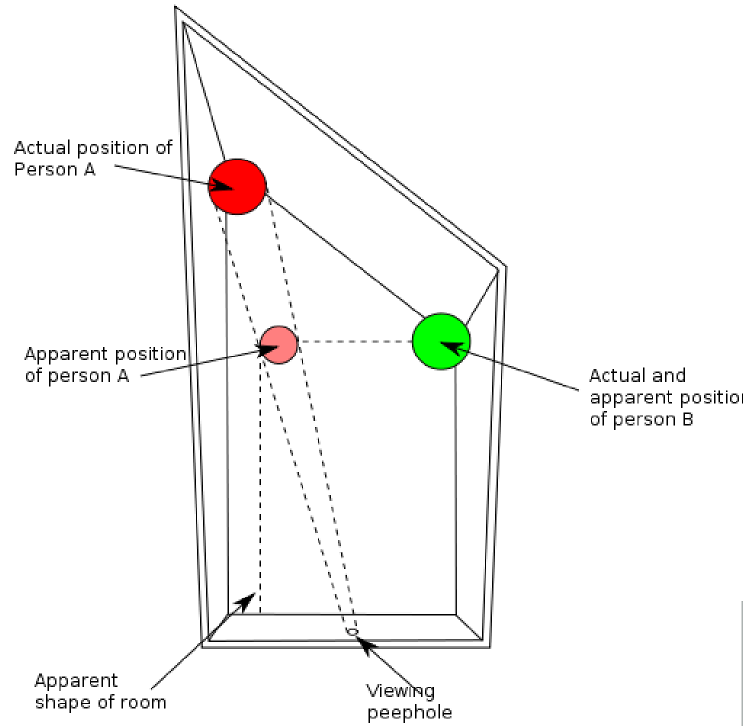
Perspective cues



Perspective cues



Ames Room

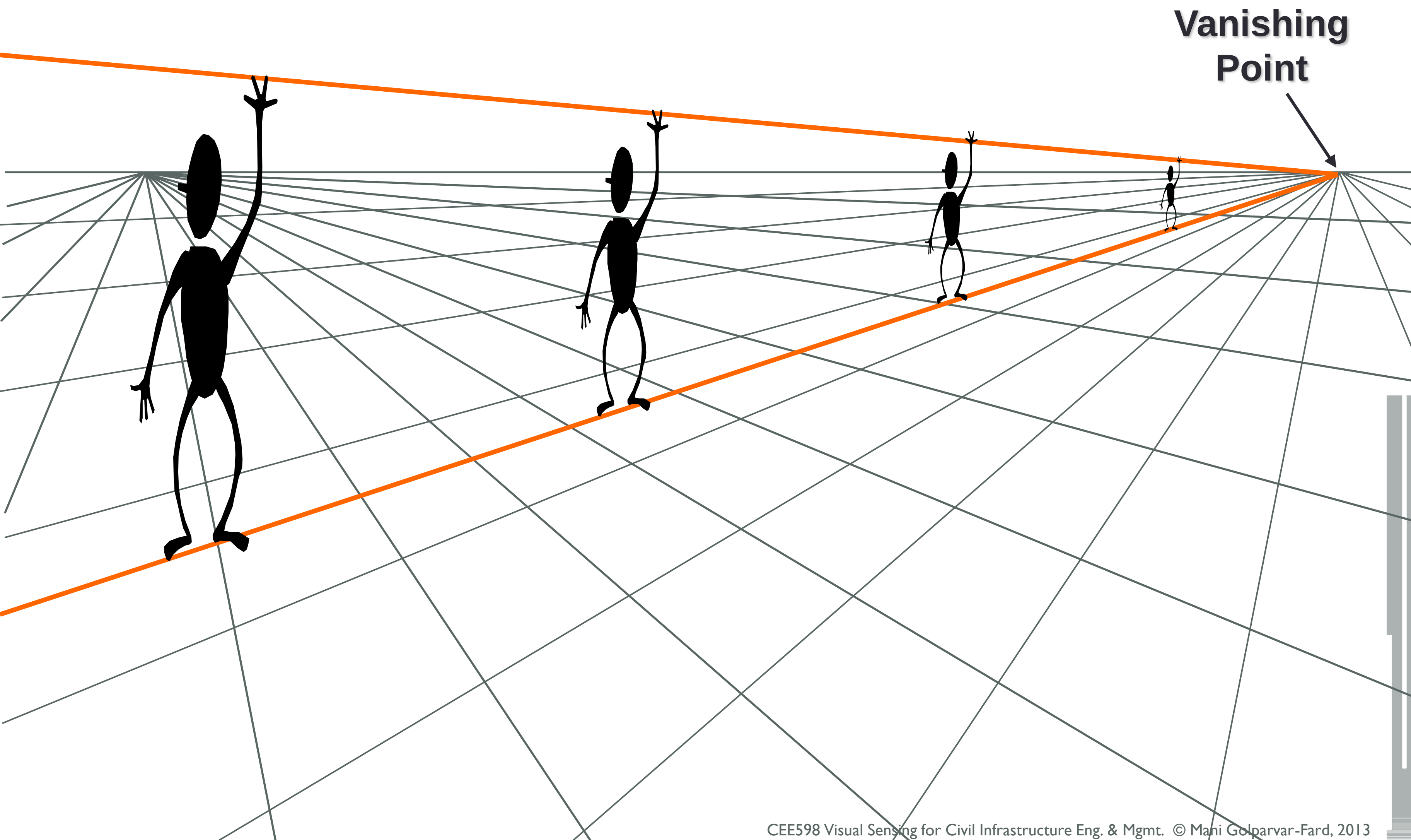


http://www.youtube.com/watch?v=6aJIX0AEWys&feature=related

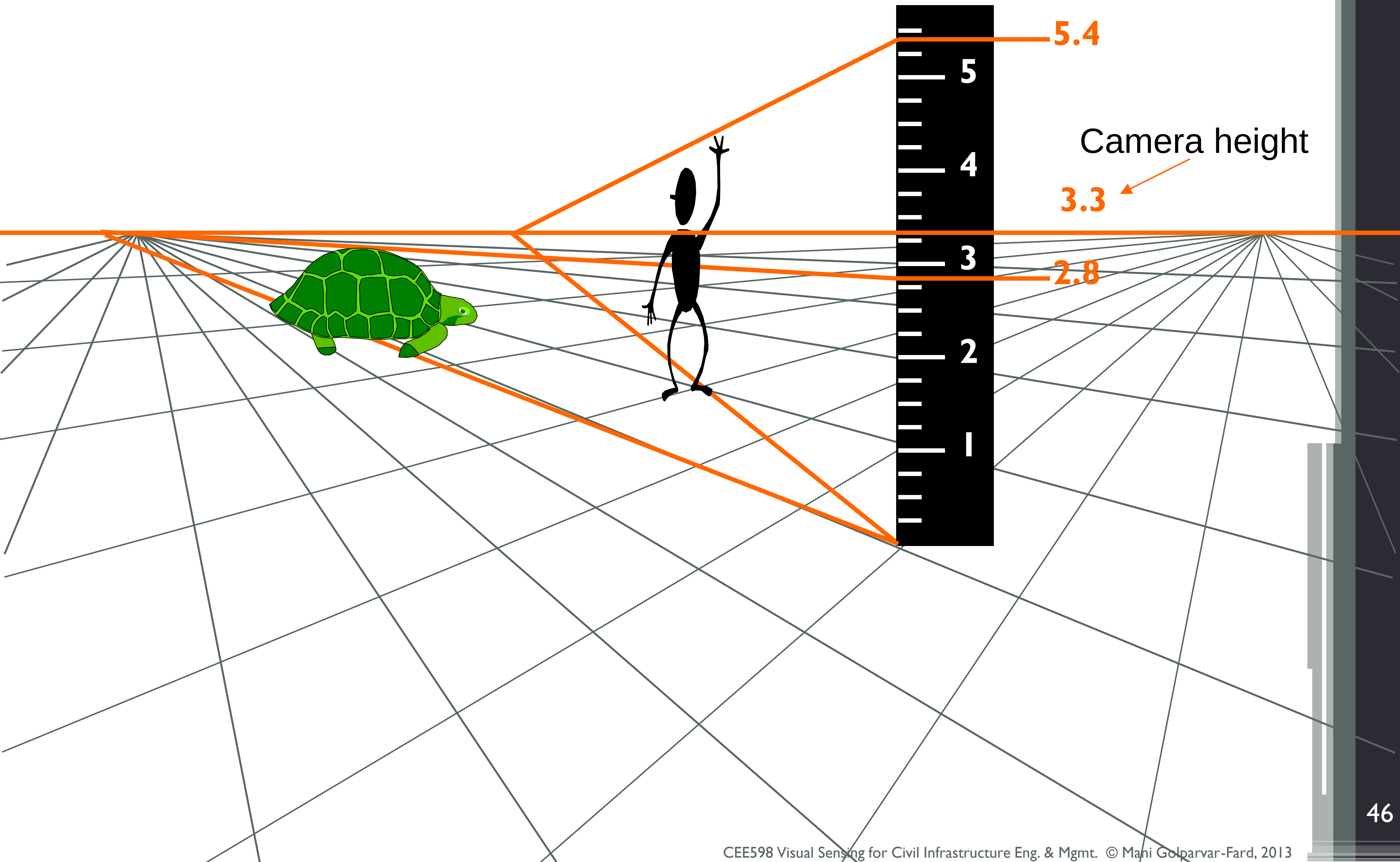
http://www.youtube.com/watch?v=hCV2Ba5wrcs&feature=related

http://www.youtube.com/watch?v=6aJIX0AEWys&feature=related

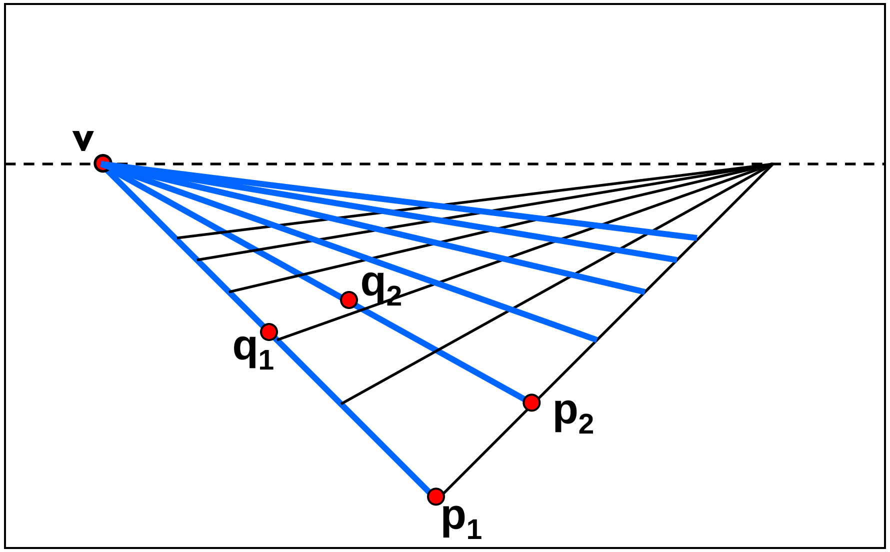
Comparing heights



Comparing heights



Computing vanishing points (from lines)



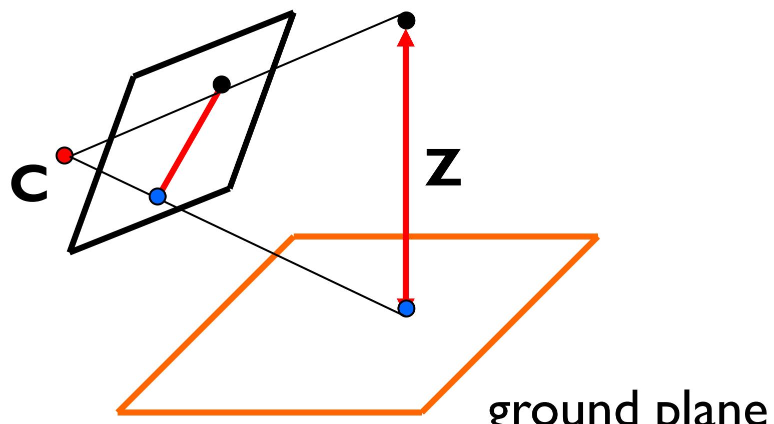
- Intersect p_1q_1 with p_2q_2

$$v = (p_1 \times q_1) \times (p_2 \times q_2)$$

Least squares version

- Better to use more than two lines and compute the "closest" point of intersection
- See notes by Bob Collins for one good way of doing this:
  - http://www-2.cs.cmu.edu/~ph/869/www/notes/vanishing.txt

Measuring height without a ruler



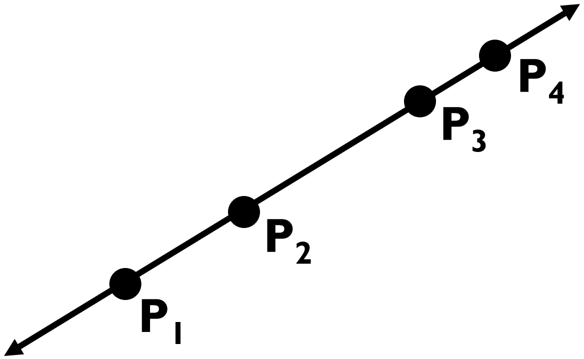
Compute Z from image measurements

- Need more than vanishing points to do this

The cross ratio

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

The cross-ratio of 4 collinear points



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

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

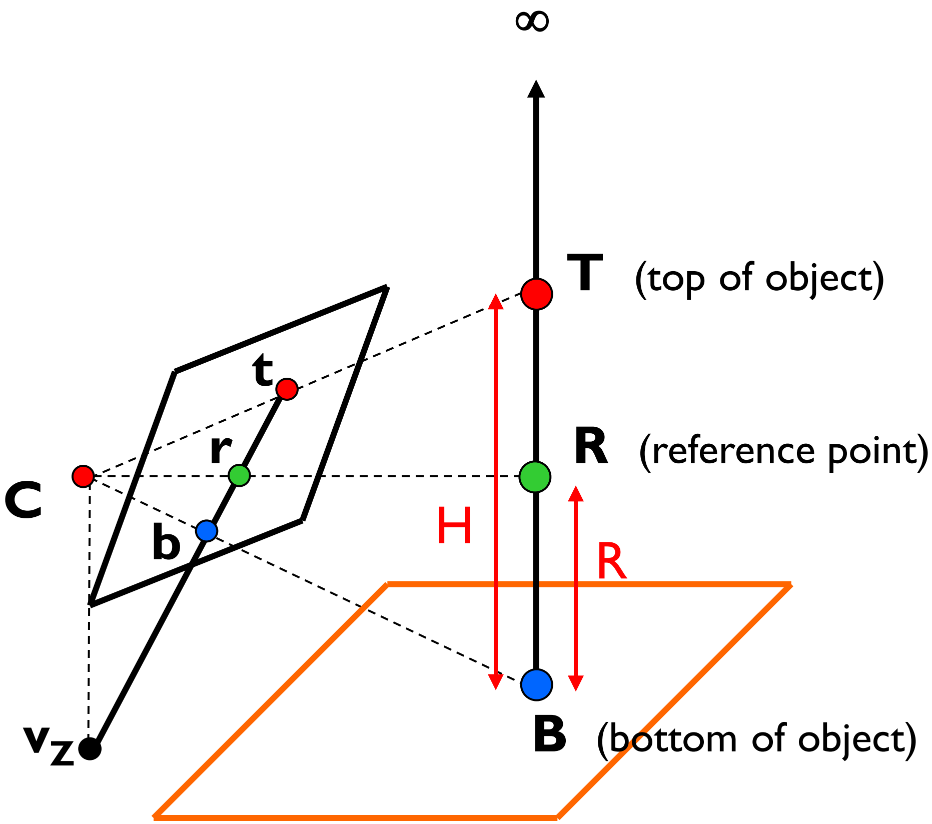
Can permute the point ordering $\frac{\|\mathbf{P}_1 - \mathbf{P}_3\| \, \|\mathbf{P}_4 - \mathbf{P}_2\|}{\|\mathbf{P}_1 - \mathbf{P}_2\| \, \|\mathbf{P}_4 - \mathbf{P}_3\|}$

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

This is the fundamental invariant of projective geometry

Proof available on Scholar

Measuring height



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

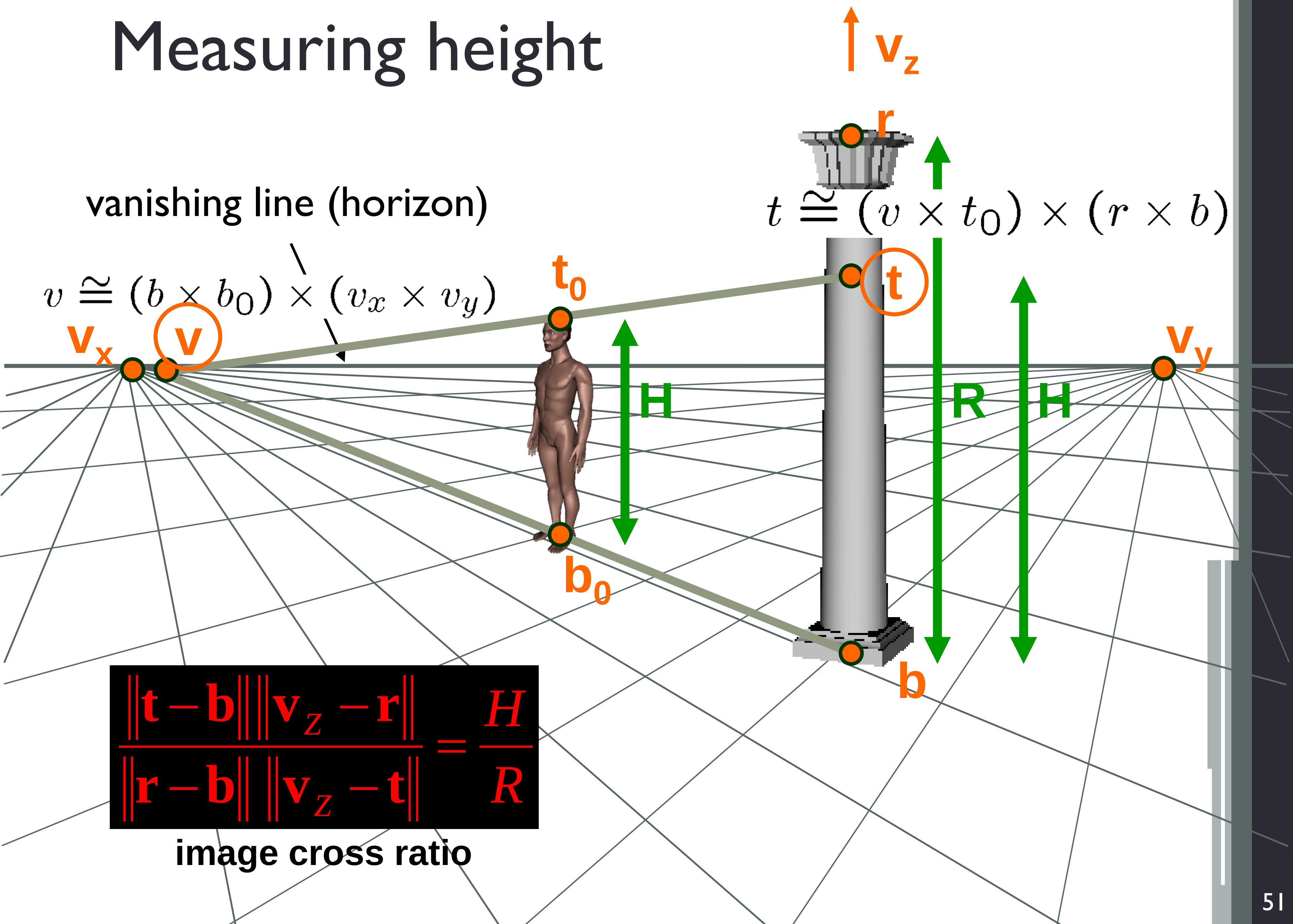
scene cross ratio

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

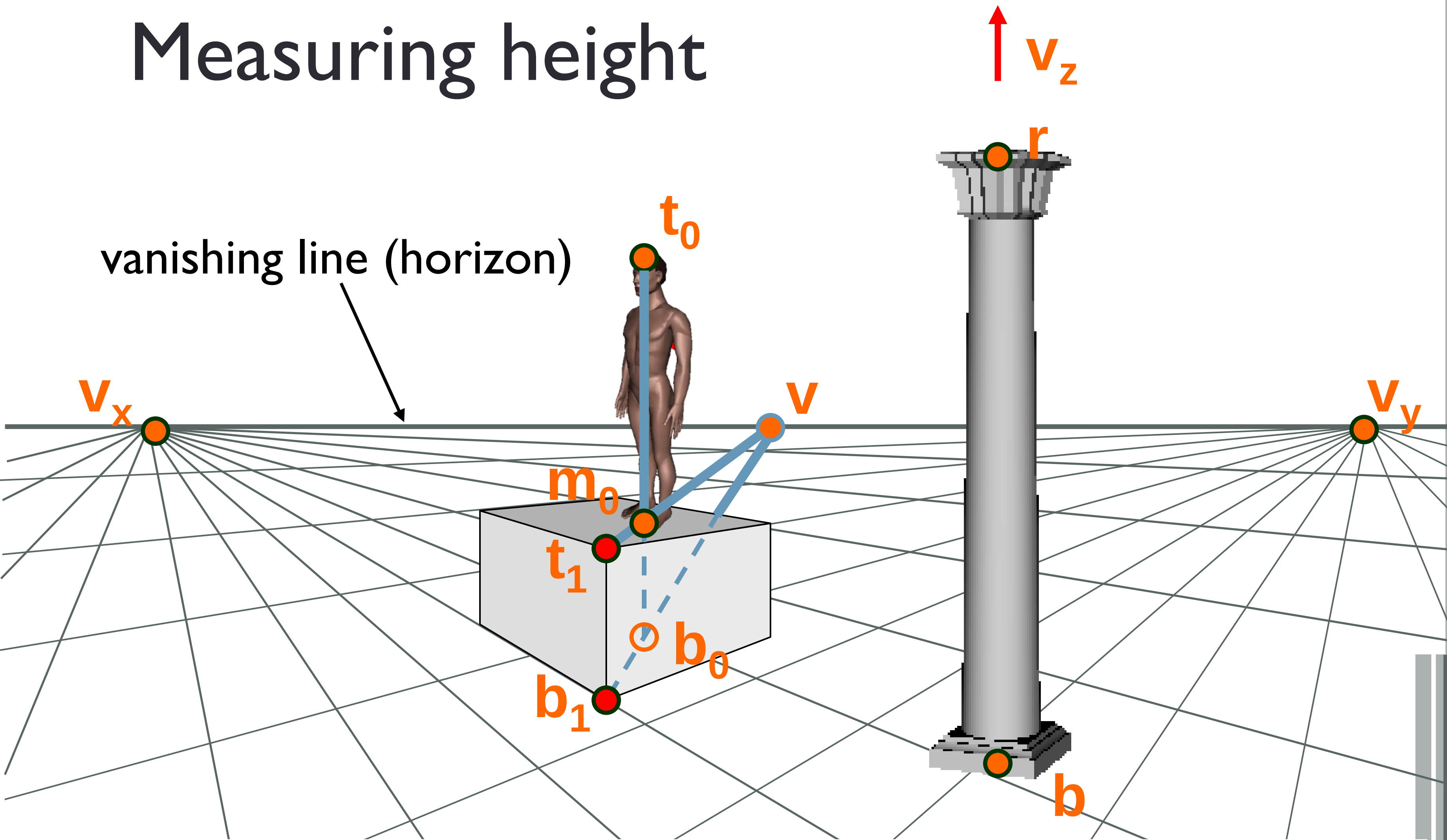
image cross ratio

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

Measuring height



Measuring height



What if the point on the ground plane $\mathbf{b_0}$ is not known?

- Here the guy is standing on the box, height of box is known
- Use one side of the box to help find $\mathbf{b_0}$ as shown above

Computing (X,Y,Z) coordinates

- Okay, we know how to compute height (Z coords)
  - how can we compute X, Y?
  - Exact same idea as before, but substitute X for Z (e.g., need a reference object with known X coordinates)

3D Modeling from a photograph

Some Related Techniques

- Image-Based Modeling and Photo Editing
  - Mok et al., SIGGRAPH 2001
  - http://graphics.csail.mit.edu/ibedit/
- Single View Modeling of Free-Form Scenes
  - Zhang et al., CVPR 2001
  - http://grail.cs.washington.edu/projects/svm/
- Tour Into The Picture
  - Anjyo et al., SIGGRAPH 1997
  - http://koigakubo.hitachi.co.jp/little/DL_TipE.html