EBU7240
Computer Vision
- Calibration -

Semester 1, 2021

Changjae Oh

Real

Simulation

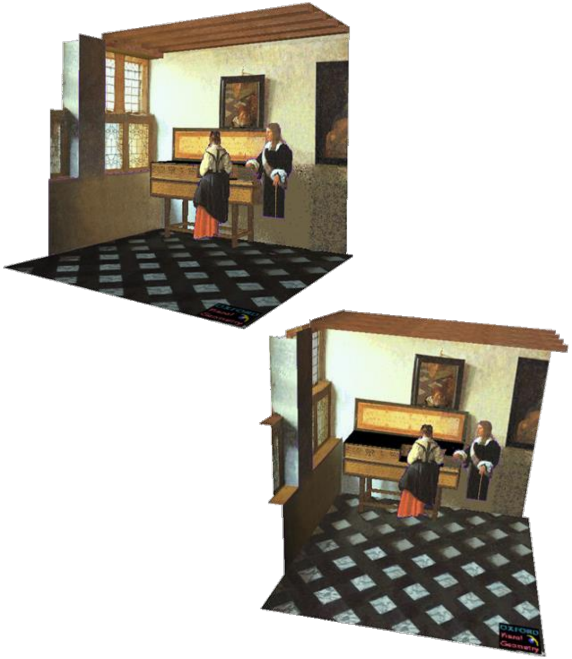
Objectives

- Understanding the **concept of camera calibration**
- Understanding the **relationship between *image coordinate*, *camera coordinate*, and *world coordinate***
- Understanding a **linear method** for camera calibration

Our goal: Recovery of 3D structure

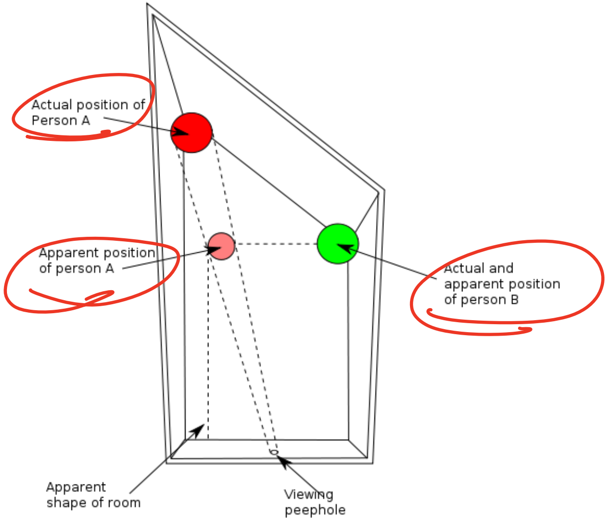
depth information

J. Vermeer, *Music Lesson*, 1662



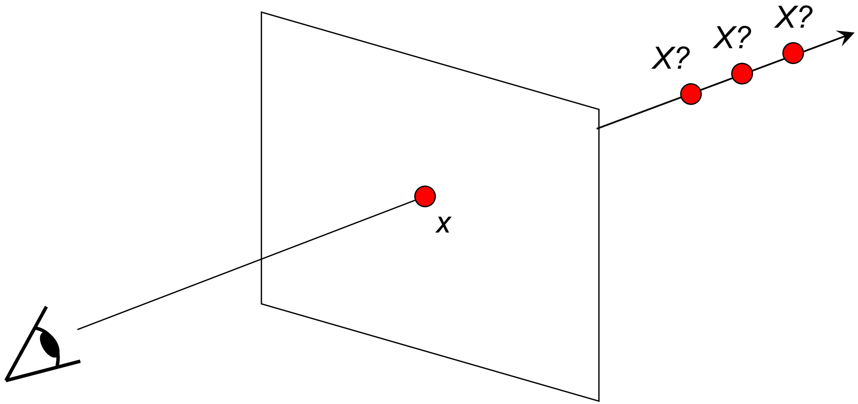
A. Criminisi, M. Kemp, and A. Zisserman, Bringing Pictorial Space to Life: computer techniques for the analysis of paintings, *Proc. Computers and the History of Art*, 2002

Things aren't always as they appear…



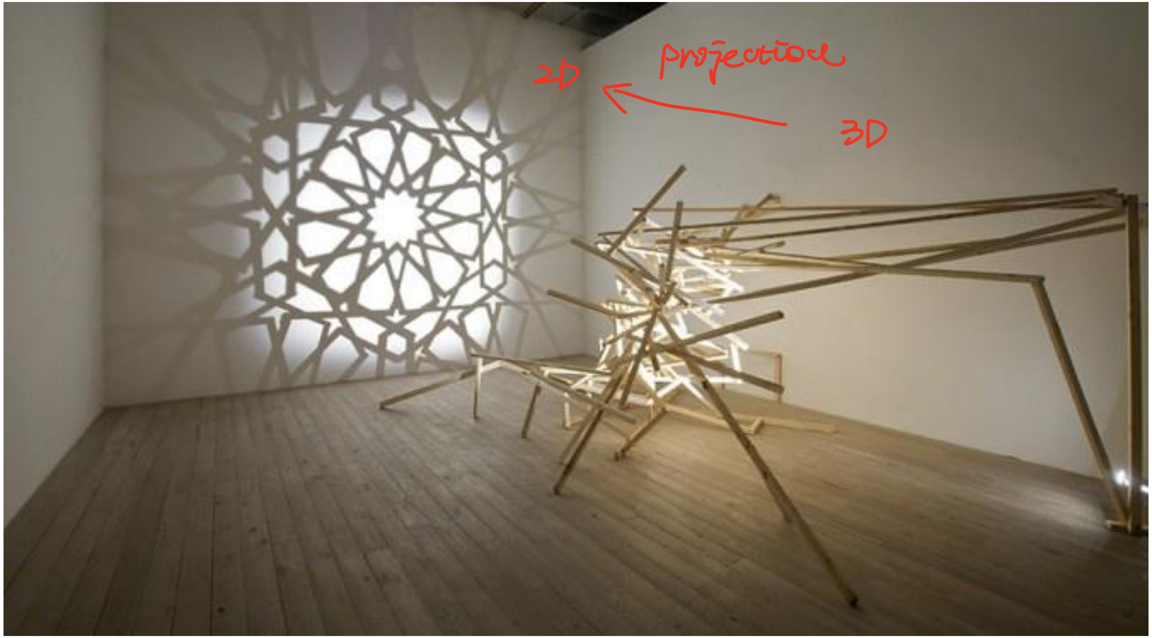
http://en.wikipedia.org/wiki/Ames_room

Single-view ambiguity



Single-view ambiguity

Single-view ambiguity

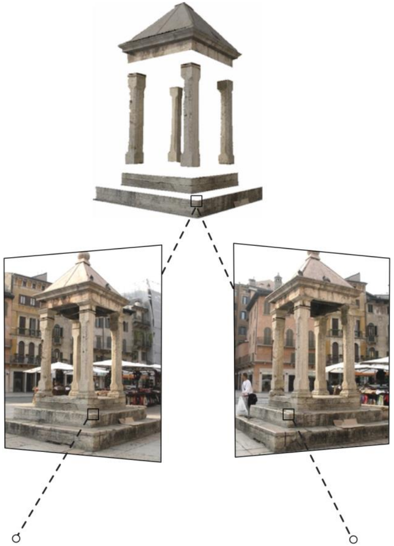


Rashad Alakbarov shadow sculptures

Our goal: Recovery of 3D structure

- **When certain assumptions hold, we ca n recover structure from a single view**

- **In general, we need *multi-view geometry***

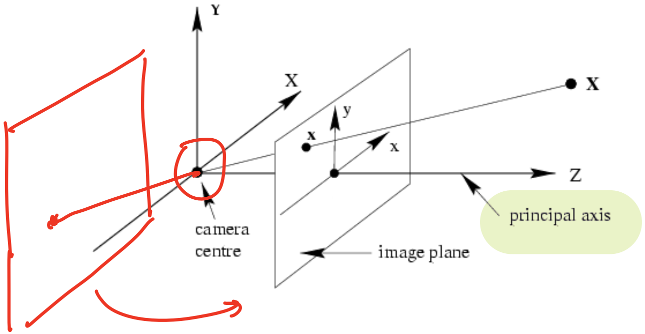


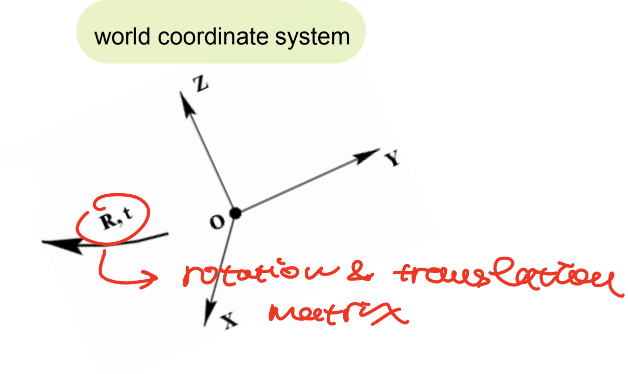
- But first, we need to understand the geometry of a single camera…

Image source

Camera calibration

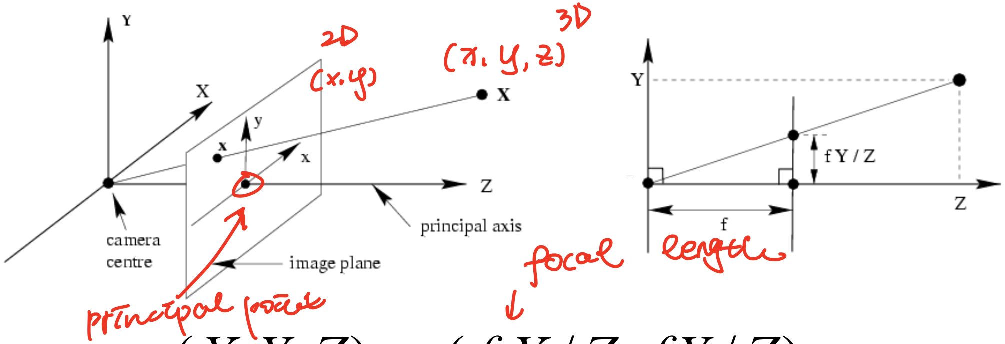
- **Camera calibration:**
  - figuring out transformation from world coordinate system to image coordinate system





- 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

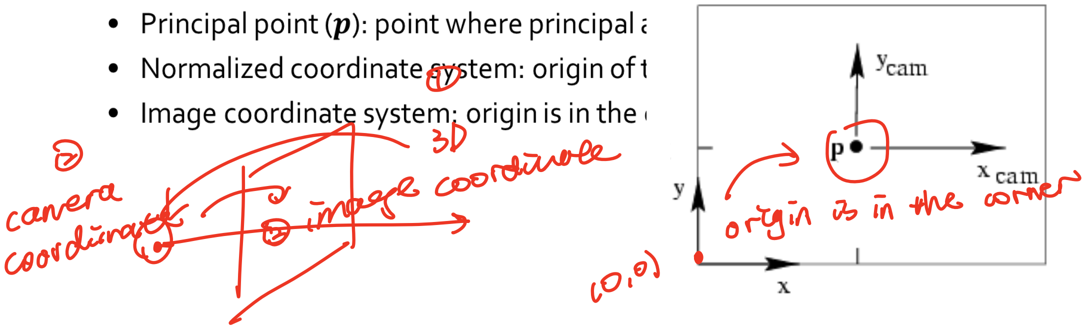
Review: Pinhole camera model



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

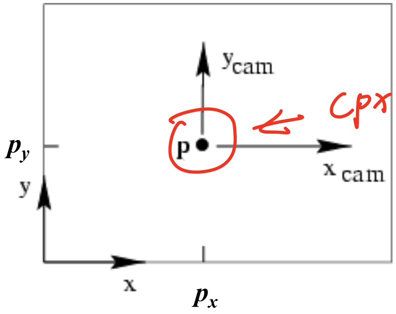
$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Principal point



- Principal point ($\boldsymbol{p}$): point where principal axis intersects the image plane
- ~~Normalized~~ camera coordinate system: origin of the image is at the principal point
- Image coordinate system: origin is in the corner

Principal point offset



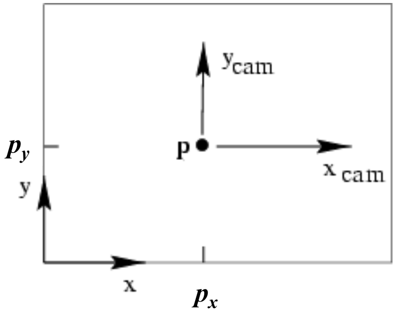
(p_x, p_y)

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

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

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

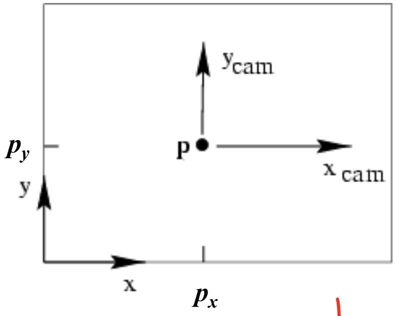
Principal point offset



principal point: (p_x, p_y)

$$\begin{bmatrix} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Principal point offset



principal point: (p_x, p_y)

$$\begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & 0 \\ & 1 & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

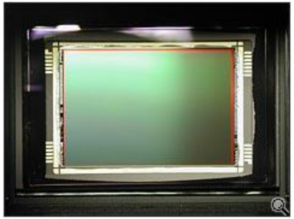
calibration matrix K   projection matrix $[\mathrm{I} \mid 0]$ (augmented matrix)

$$\underbrace{\mathrm{K} \qquad [\mathrm{I} \mid 0]}$$

$$\mathrm{P} = \mathrm{K}[\mathrm{I} \mid 0]$$

Pixel coordinates

3D → camera → principal offset



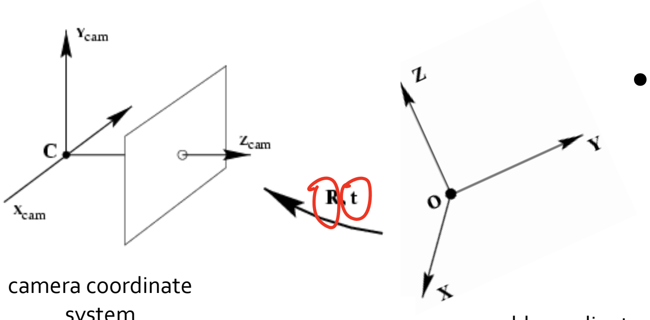
Pixel size: $\dfrac{1}{m_x} \times \dfrac{1}{m_y}$

- 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



camera coordinate system

world coordinate system

- 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):

rotation matrix.

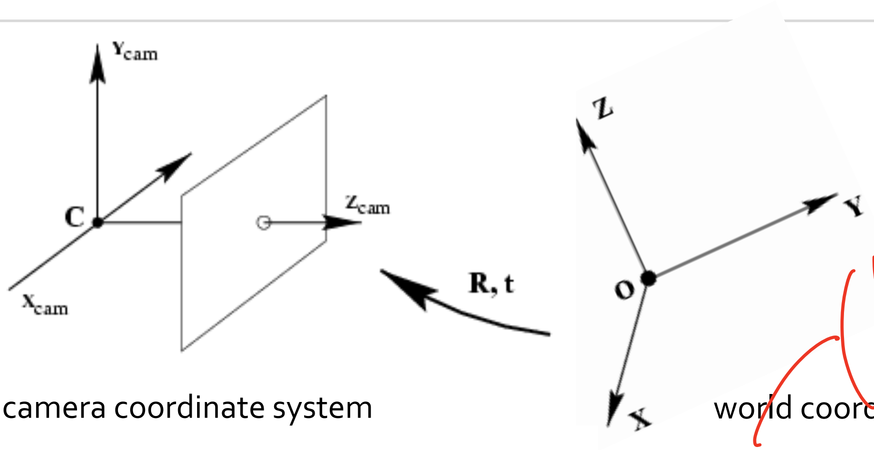
$$\tilde{X}_{cam} = R(\tilde{X} - \tilde{C})$$

coords. of point in camera frame

coords. of a point in world frame

coords. of camera center in world frame

Camera rotation and translation



camera coordinate system

world coordinate system

3x3

$$\begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

3x3

$$\begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$$

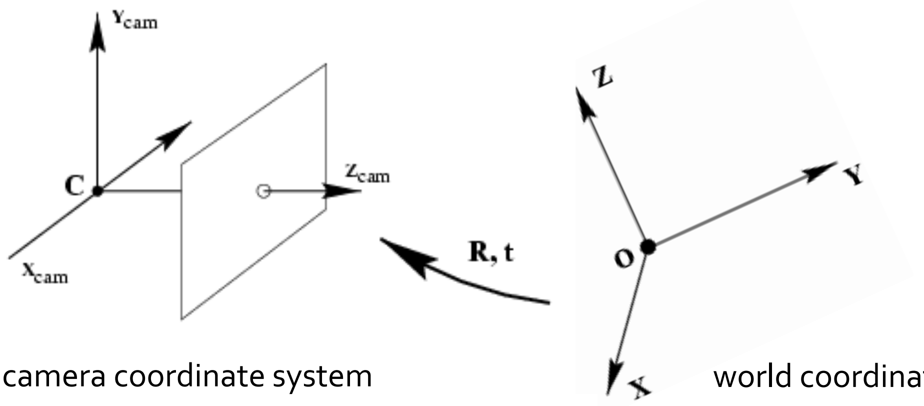
3x1

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

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

3D transformation matrix (4 x 4)

Camera rotation and translation



camera coordinate system

world coordinate system

camera coordinate

world coordinate

$\begin{bmatrix} X \\ 1 \end{bmatrix}$

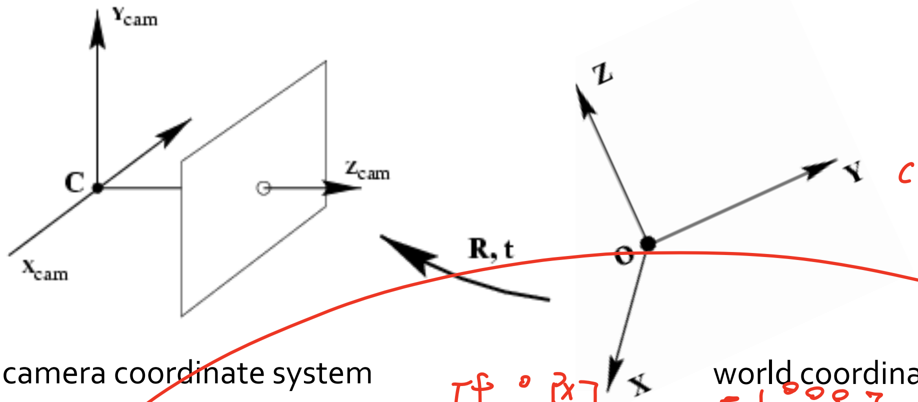
$\begin{bmatrix} X \\ 1 \end{bmatrix}$

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

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

3D transformation matrix (4 x 4)

Camera rotation and translation



camera coordinate system

world coordinate system

camera calibration

$$\begin{bmatrix} f & 0 & px \\ 0 & f & py \\ 0 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

2D-point in image coordinate D{3x1}

$$\mathrm{x} = \mathrm{K}[\mathrm{I} \mid 0] \begin{bmatrix} \mathrm{R} & -\mathrm{R}\widetilde{\mathrm{C}} \\ 0 & 1 \end{bmatrix} \mathrm{X}$$

3D-point in world coordinate

D{4x1}

calibration matrix

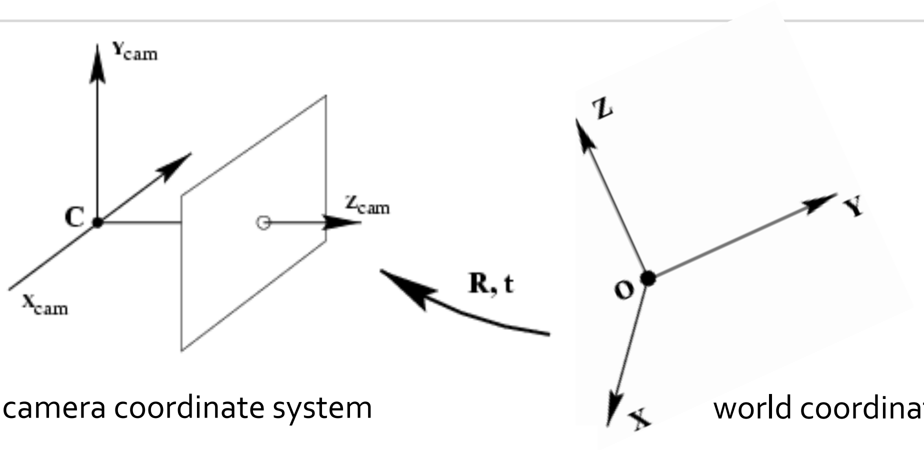
2D transformatio
n matrix (3 x 3)

perspective project
ion matrix (3 x 4)

3D transformation
matrix (4 x 4)

D{3x4}

Camera rotation and translation

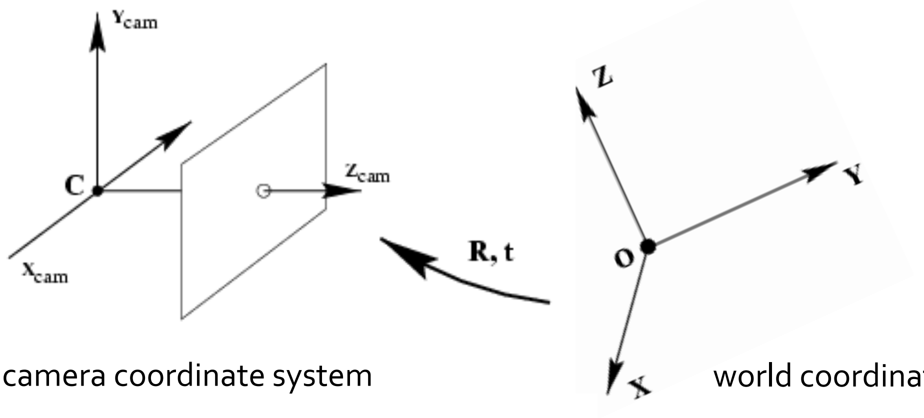


camera coordinate system

world coordinate system

$$\mathbf{x} = K\left[R \mid -R\tilde{C}\right]X$$

Camera rotation and translation



camera coordinate system

world coordinate system $\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$

img ↔ cam   cam ↔ world

$\tilde{C}$ ⇒ coordinates of camera center

$$x = K[R \mid t]X \qquad t = -R\tilde{C}$$

R ⇒ 3D rotation matrix

$$K = \begin{bmatrix} m_x & & \\ & m_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix}$$

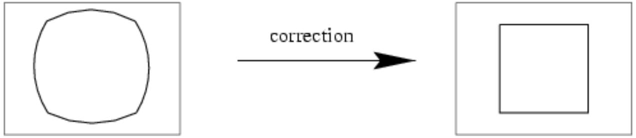
Camera parameters $\mathbf{P} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]$

- **Intrinsic parameters**
  - Principal point coordinates $(p_x \; p_y)$
  - Focal length f
  - Pixel magnification factors $(m_x \; m_y)$
  - *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



Camera parameters $\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}$

- **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

coords. of camera center in world frame

$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}}\end{bmatrix}$$

  - What is the projection of the camera center?

$$\mathbf{PC} = \mathbf{K}\begin{bmatrix}\mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}}\end{bmatrix}\begin{bmatrix}\tilde{\mathbf{C}} \\ 1\end{bmatrix} = 0$$

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

Camera calibration

$$\lambda \mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

3x4

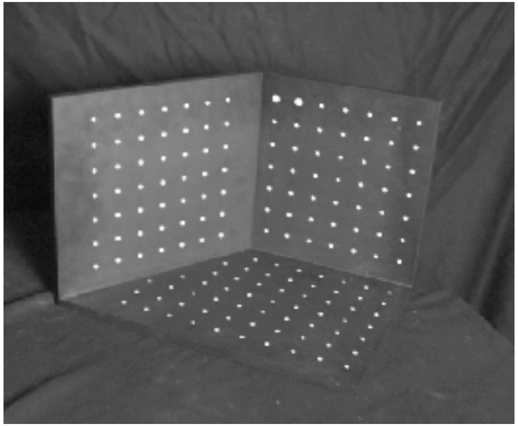
$$\begin{bmatrix}\lambda x \\ \lambda y \\ \lambda\end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix}\begin{bmatrix}X \\ Y \\ Z \\ 1\end{bmatrix}$$

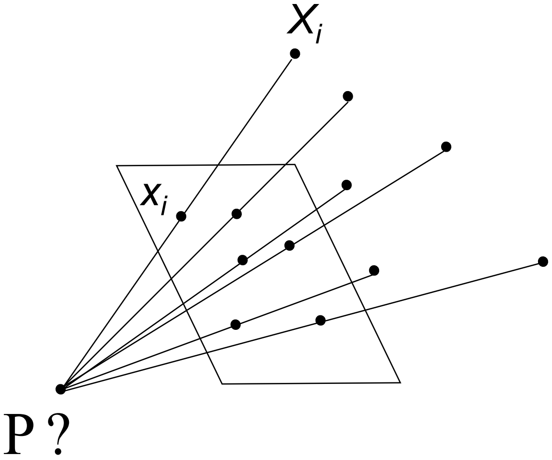
1

11 unknown variables

Camera calibration

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





Camera calibration: Linear method

- $\boldsymbol{P}$ has 11 degrees of freedom
- One 2D/3D correspondence gives us two linearly independent equations
  - 6 correspondences needed for a minimal solution

$$\begin{bmatrix} \lambda x \\ \lambda y \\ \lambda \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

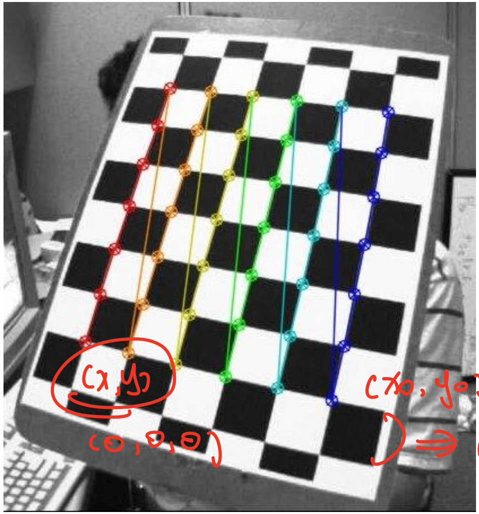
Camera calibration: Linear method

- P has 11 degrees of freedom
- One 2D/3D correspondence gives us two linearly independent equations
  - 6 correspondences needed for a minimal solution

affine matrix

$Mx = b$,

$\begin{pmatrix} a & b & c \\ d & e & f \end{pmatrix}$ RANSAC



(x0, y0)

⇒ relative 3D point.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

3D→2D

Recall: Week1 quiz

8 equations.

- Given two point sets:
  - $\boldsymbol{x} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_4\} = \{(u_1, v_1), \dots, (u_4, v_4)\} = \{(0{,}260), (640{,}260), (0{,}400), (640{,}400)\}$
  - $\boldsymbol{x}' = \{\boldsymbol{x}'_1, \dots, \boldsymbol{x}'_4\} = \{(u'_1, v'_1), \dots, (u'_4, v'_4)\} = \{(0, 0), (400, 0), (0{,}640), (400{,}640)\}$

  Find the perspective projection matrix $\boldsymbol{P}$ such that $\boldsymbol{x}' = \boldsymbol{P}\boldsymbol{x}$

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Camera calibration: Linear method

- Directly estimate 11 unknowns in the **P** matrix using known 3D points (X, Y, Z) and measured (X_i, Y_i, Z_i) and measured feature positions (u_i, v_i)

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

$$u_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1) = m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}$$

$$v_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1) = m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}$$

Camera calibration: Linear method

- Directly estimate 11 unknowns in the **P** matrix using known 3D points (X, Y, Z) and measured (X_i, Y_i, Z_i) and measured feature positions (u_i, v_i)

6 correspondings

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

1

Camera calibration: Linear method

- Solve for Projection Matrix $\boldsymbol{P}$ using least-square techniques

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

Camera calibration: linear vs. nonlinear

- **Linear calibration is easy to formulate and solve, but it doesn't directly tell us the camera parameters**

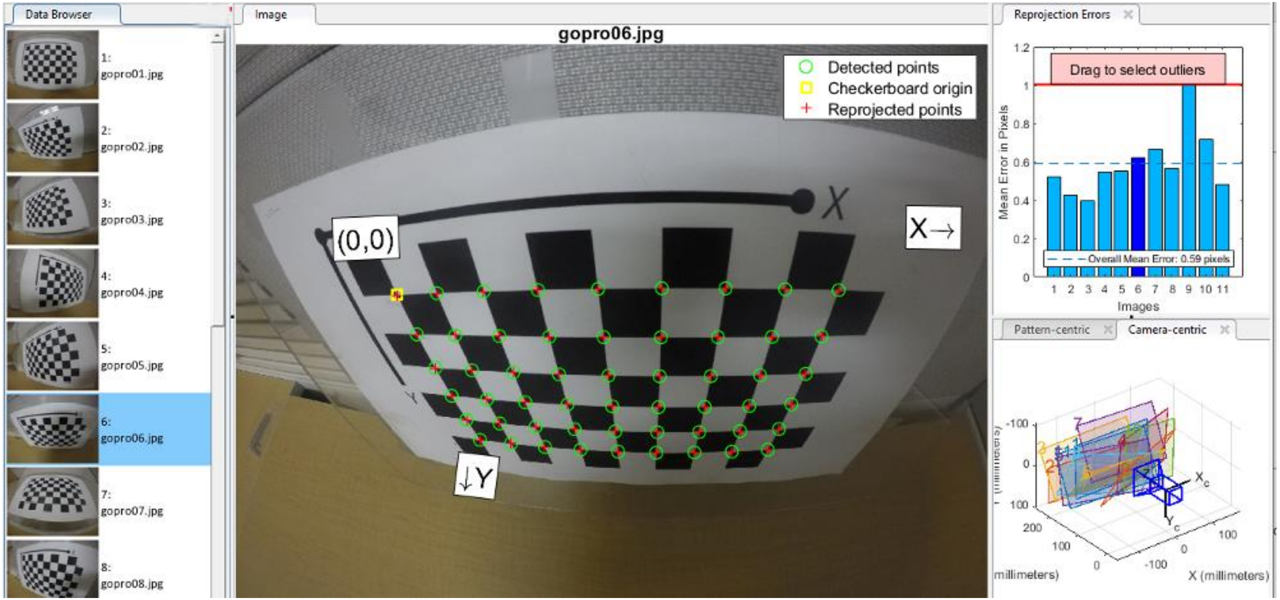
camera property

$$\begin{bmatrix} \lambda x \\ \lambda y \\ \lambda \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad \text{vs.} \quad \mathbf{x} = \mathbf{K}\begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}\mathbf{X}$$

- **In practice, non-linear methods are preferred**
  - Write down objective function in terms of intrinsic and extrinsic parameters
  - 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
  - Can model radial distortion and impose constraints such as known focal length and or thogonality

Application?

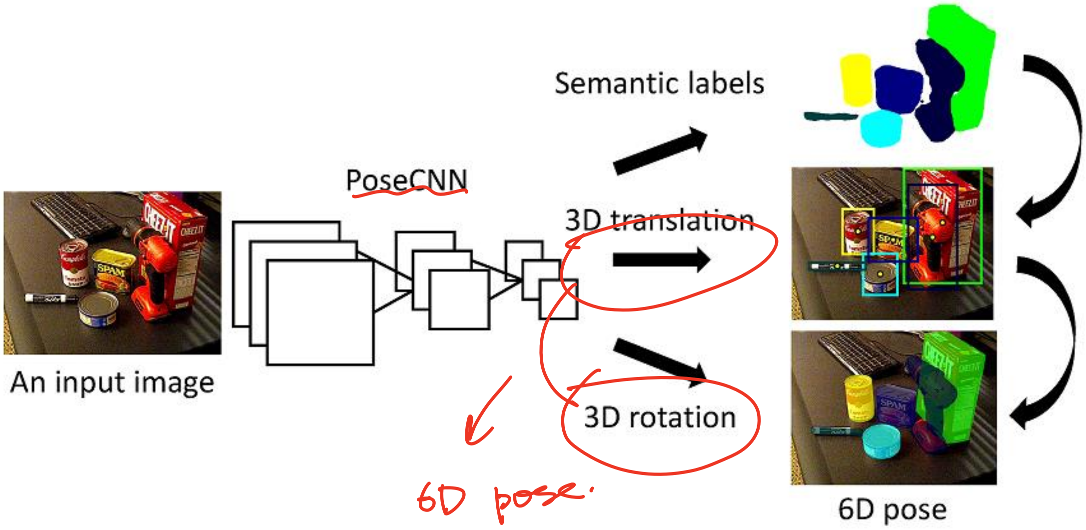
- **Calibration is fundamental task for various computer vision tasks**



https://ch.mathworks.com/help/vision/ug/single-camera-calibrator-app.html

Application?

- **Calibration is fundamental task for various computer vision tasks**



Xiang et al. "PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes." RSS, 2018

EBU7240
Computer Vision
- Single-view Modeling -

Semester 1, 2021

Changjae Oh

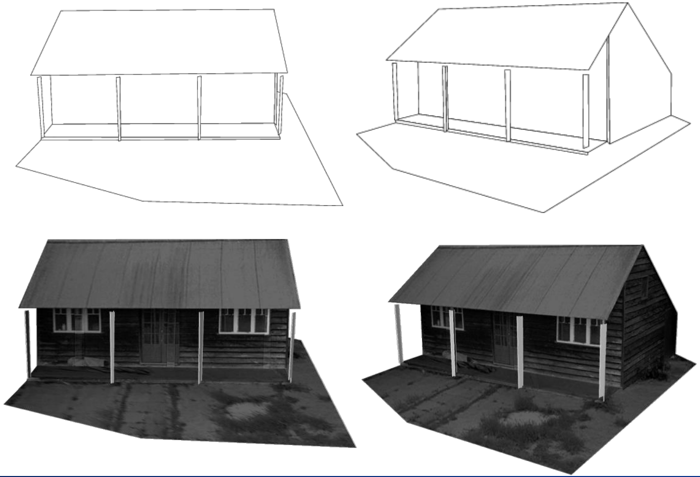
Objectives

- To understand calibration from vanishing points ⇒ 2D → 3D
- To understand measuring height without ruler

Application: Single-view modelling

✓

2D → 3D



A. Criminisi, I. Reid, and A. Zisserman, [Single View Metrology](), IJCV 2000

Camera calibration revisited

Linear caliberation method
6 corresponding point pairs

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

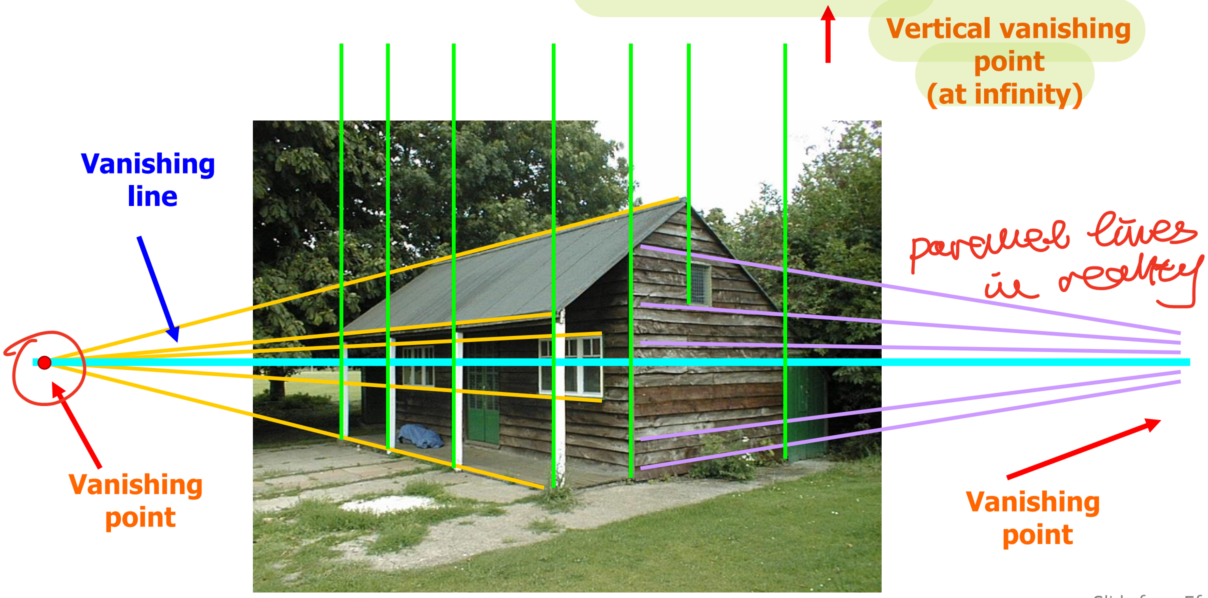
2D → 3D

(x, y)

(X, Y, Z)

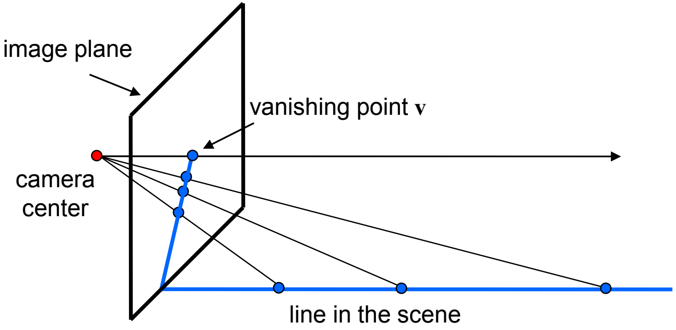
Camera calibration revisited

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



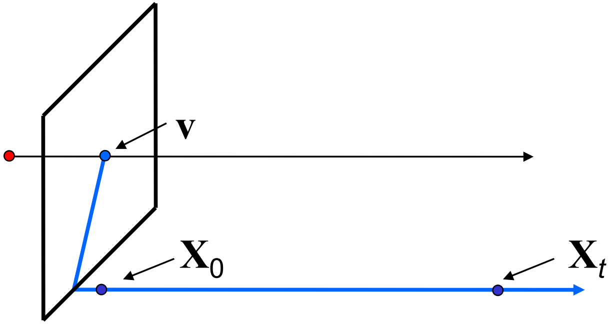
Slide from Efros, Photo from Criminisi

Recall: Vanishing points



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

Computing vanishing points

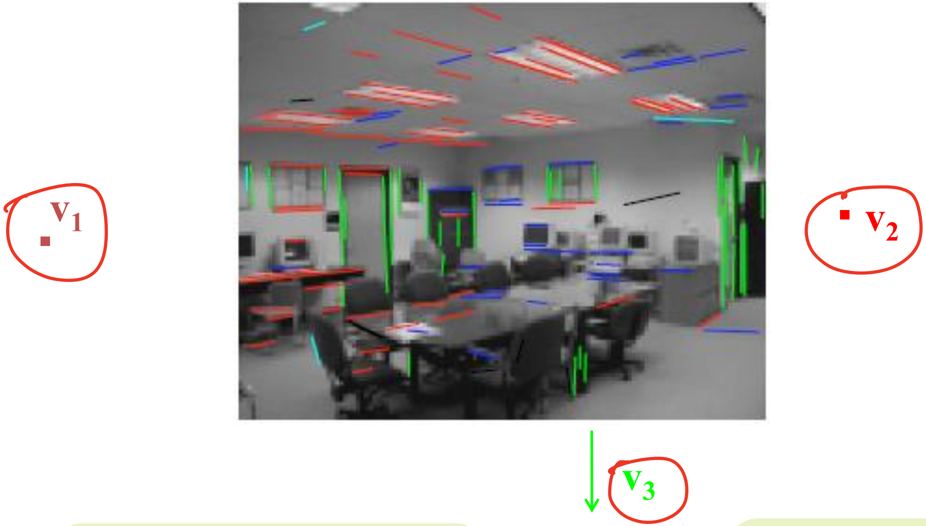


$$\mathbf{X}_t = \begin{bmatrix} x_0 + td_1 \\ y_0 + td_2 \\ z_0 + td_3 \\ 1 \end{bmatrix} = \begin{bmatrix} x_0 / t + d_1 \\ y_0 / t + d_2 \\ z_0 / t + d_3 \\ 1/t \end{bmatrix} \qquad \lim_{t\to\infty} \mathbf{X}_\infty = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ 0 \end{bmatrix} \Rightarrow$$

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

Calibration from vanishing points

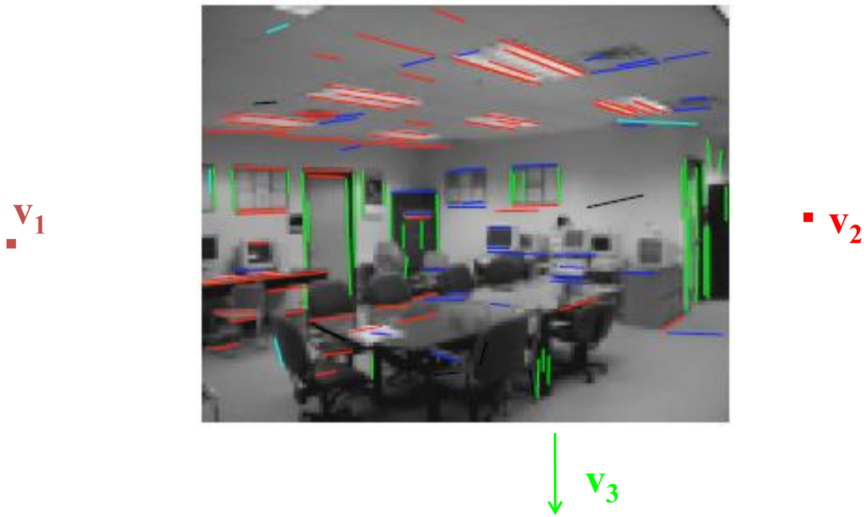
- Consider a scene with three orthogonal vanishing directions:



- Note: v1, v2 are finite vanishing points and v3 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

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

- $p_1 = P(1,0,0,0)^T$ – the vanishing point in the x direction
- Similarly, p_2 and p_3 are the vanishing points in the y and z directions
- $p_4 = P(0,0,0,1)^T$ – projection of the origin of the world coordinate system
- Problem: we can only know the four columns up to independent scale factors, addition al 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:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

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

Calibration from vanishing points

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

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\lambda_i \mathbf{v}_i = \mathbf{K}\mathbf{R}\mathbf{e}_i$$

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

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

~ rotation matrix

$R \cdot R^T = I, \ (R^T = R^{-1})$

- Orthogonality constraint: $\mathbf{e}_i^T \mathbf{e}_j = 0$

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

Calibration from vanishing points

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

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\lambda_i \mathbf{v}_i = \mathbf{K}\mathbf{R}\mathbf{e}_i$$

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

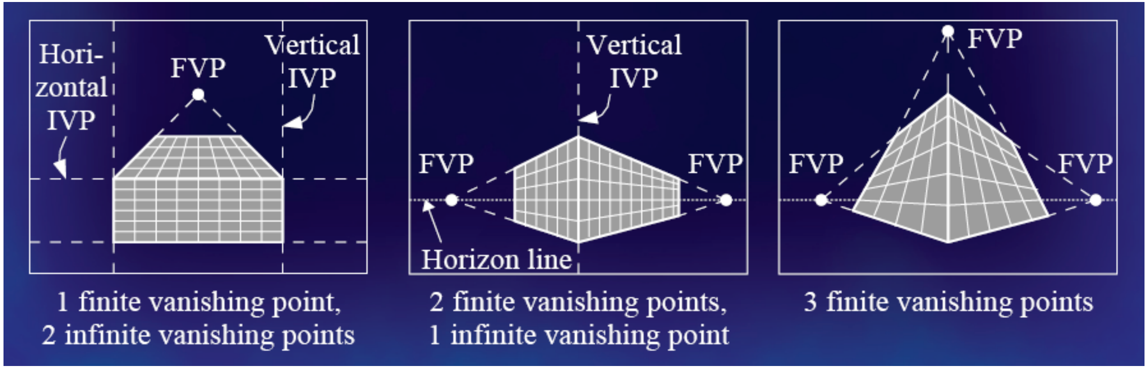
- Orthogonality constraint: $\mathbf{e}_i^T \mathbf{e}_j = 0$

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

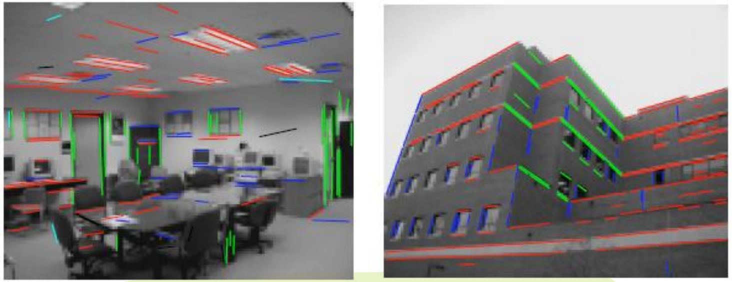
focal length: f
principle point

- Rotation disappears, each pair of vanishing points gives constraint on focal length and principal point

Calibration from vanishing points



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



Can solve for focal length, principal point

Rotation from vanishing points

intrinsic

- Constraints on vanishing points: $\lambda_i \mathbf{v}_i = \mathbf{K}\mathbf{R}\mathbf{e}_i$

- After solving for the calibration matrix: $\lambda_i \mathbf{K}^{-1}\mathbf{v}_i = \mathbf{R}\mathbf{e}_i$

- Notice: $\mathbf{R}\mathbf{e}_1 = [\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{r}_3]\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \mathbf{r}_1$

$Re_0 = r_0$

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

$r_1 = \begin{bmatrix} \cos\theta \\ -\sin\theta \\ 0 \end{bmatrix} \quad r_2 = \begin{bmatrix} \sin\theta \\ \cos\theta \\ 0 \end{bmatrix} \quad r_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

- Get λ_i by using the constraint $\|r_i\|^2 = 1$.

$\|r_1\|^2 = \|r_2\|^2 = \|r_3\|^2 = 1$

Calibration from vanishing points: Summary

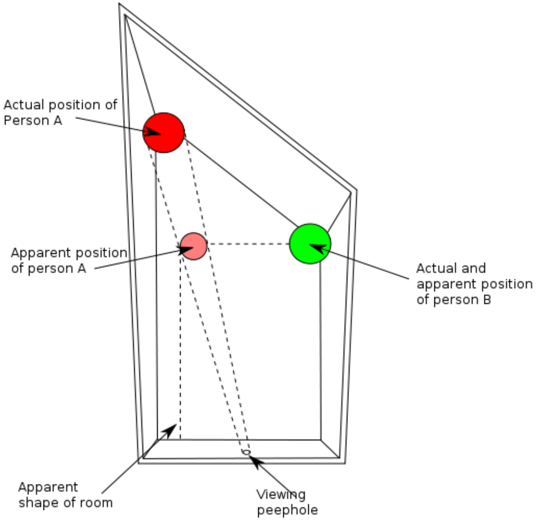
- Solve for K (focal length, principal point) using three orthogonal vanishing points
- Get rotation directly from vanishing points once calibration matrix is known

↑

- Advantages
  - No need for calibration chart, 2D-3D correspondences

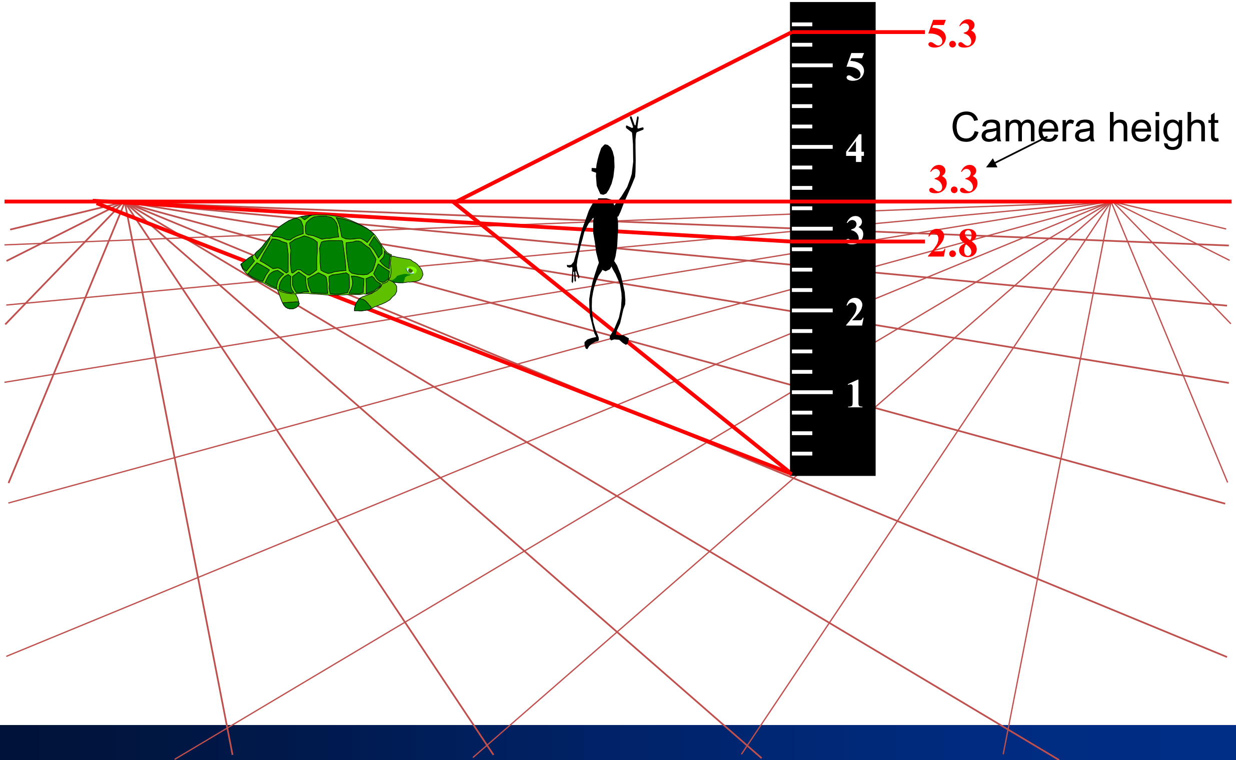
    ① calibration chart
  - Could be completely automatic
- Disadvantages
  - Only applies to certain kinds of scenes (if there is only 1 finite vanishing point)
  - Inaccuracies in computation of vanishing points
  - Problems due to infinite vanishing points

Making measurements from a single image

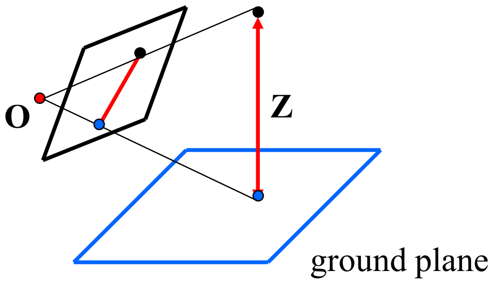


http://en.wikipedia.org/wiki/Ames_room

Recall: Measuring height



Measuring height without a ruler



Compute Z from image measurements

- Need more than vanishing points to do this

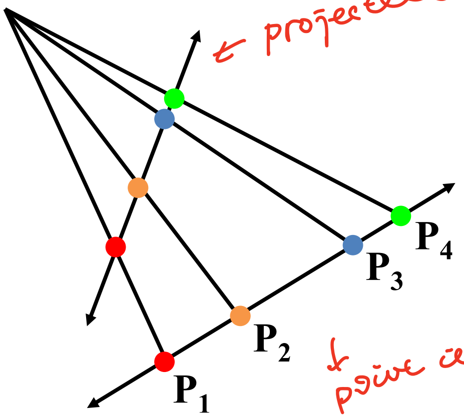
Projective invariant

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

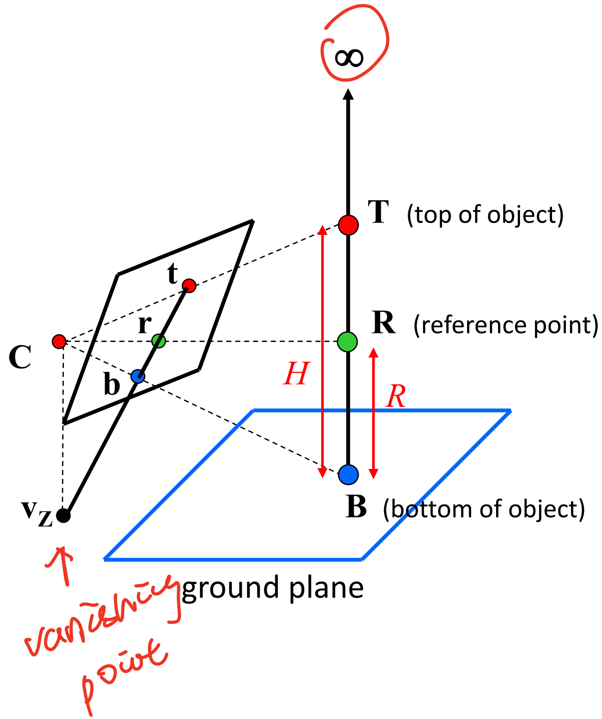
← projected co-ordinates



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

← point in real world / world coordinates

Measuring height



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

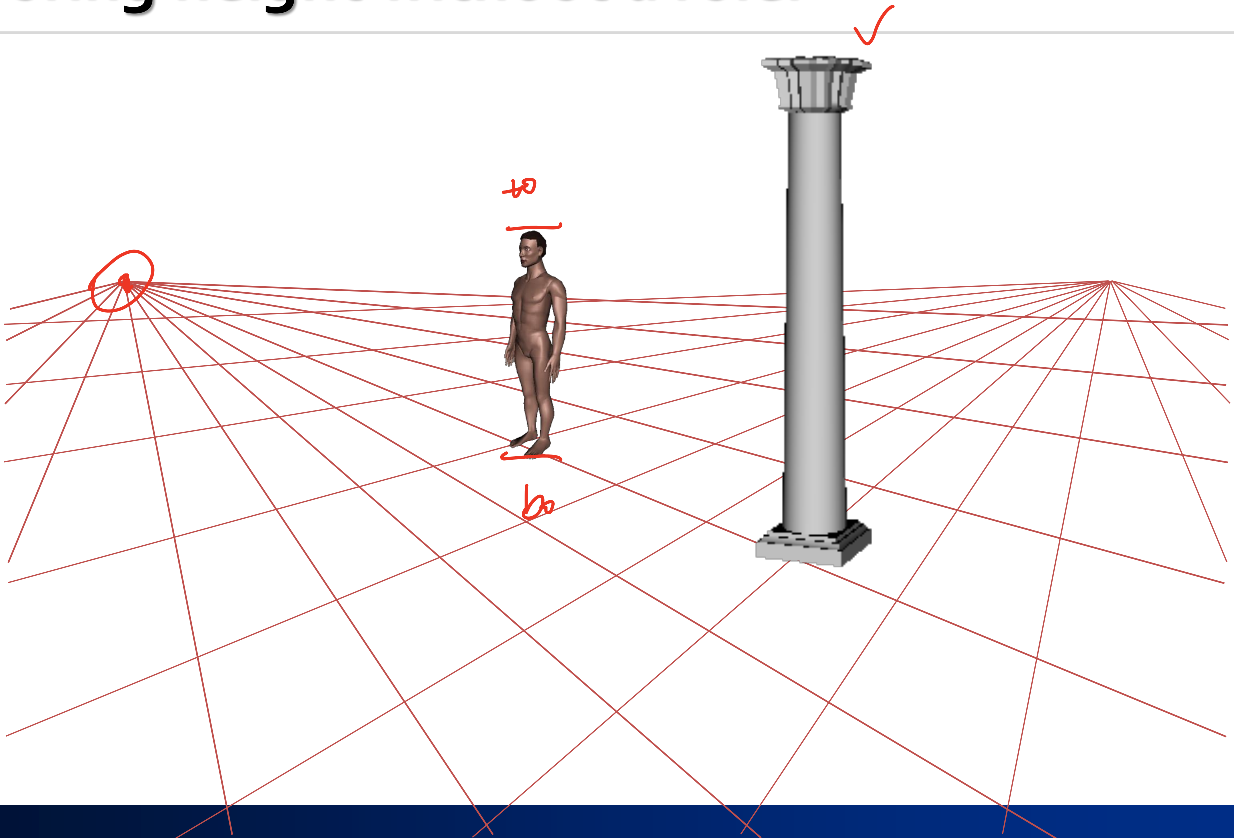
scene cross ratio

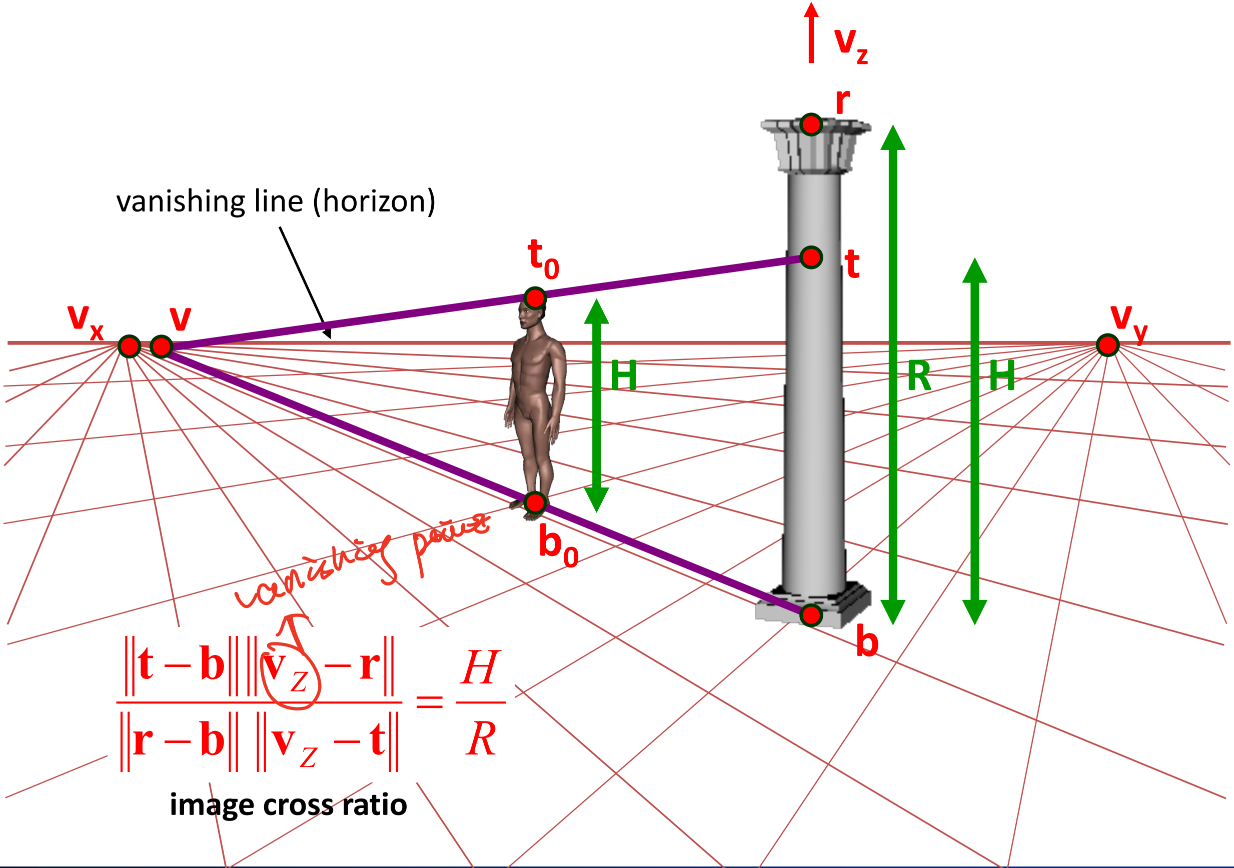
project invariance property

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

image cross ratio

Measuring height without a ruler

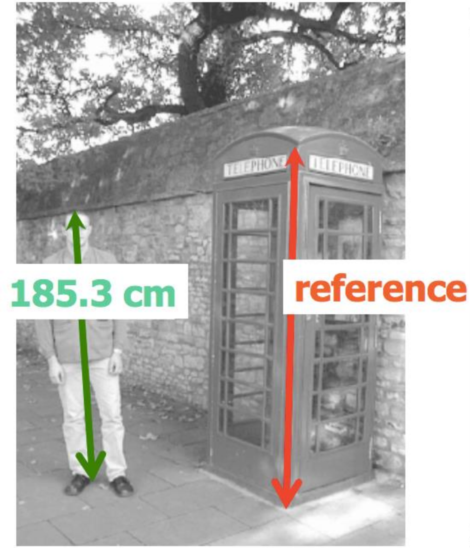


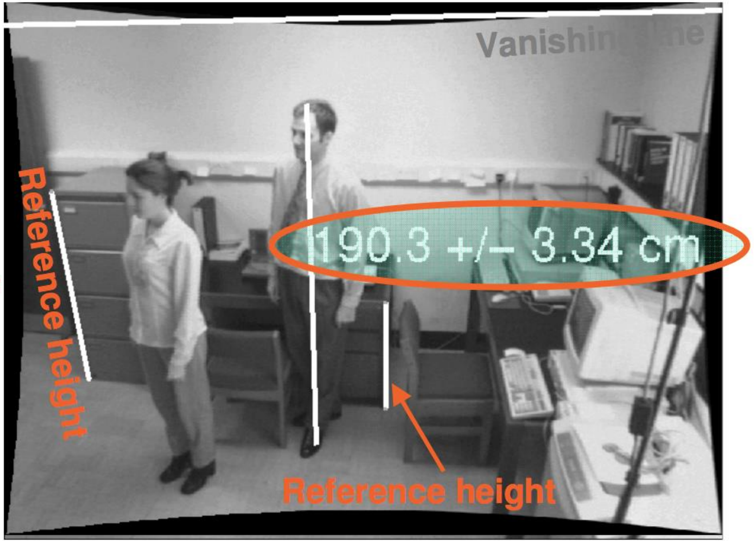


$$\frac{\|\mathbf{t}-\mathbf{b}\|\,\|\mathbf{v}_Z-\mathbf{r}\|}{\|\mathbf{r}-\mathbf{b}\|\,\|\mathbf{v}_Z-\mathbf{t}\|}=\frac{H}{R}$$

image cross ratio

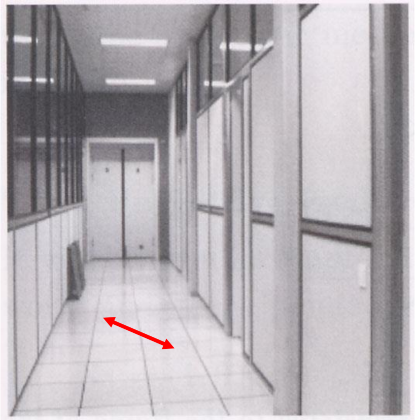
Examples

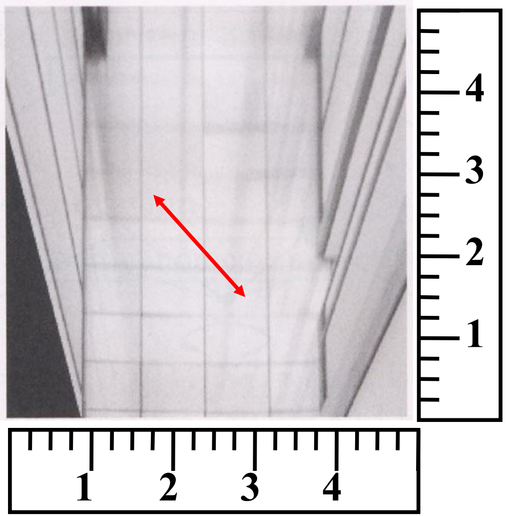




A. Criminisi, I. Reid, and A. Zisserman, Single View Metrology, IJCV 2000
Figure from UPenn CIS580 slides

Measurements on planes

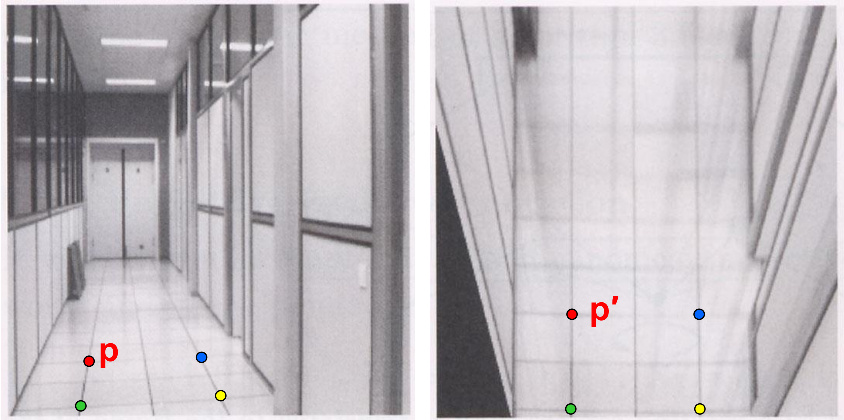




Approach: unwarp then measure

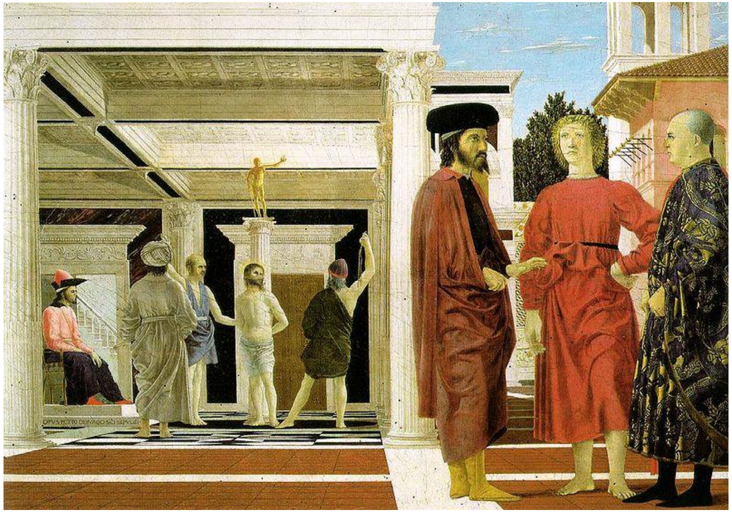
What kind of warp is this?

Image rectification



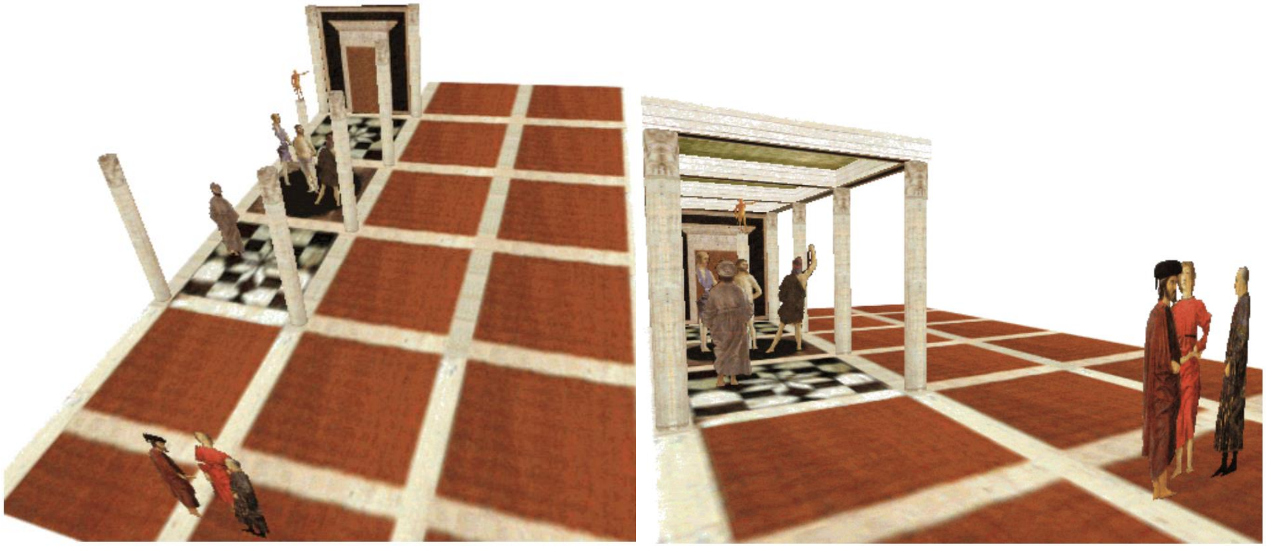
- **To unwarp (rectify) an image**
  - solve for homography H given p and p'
  - how many points are necessary to solve for H?

Image rectification: example



Piero della Francesca, *Flagellation,* ca. 1455

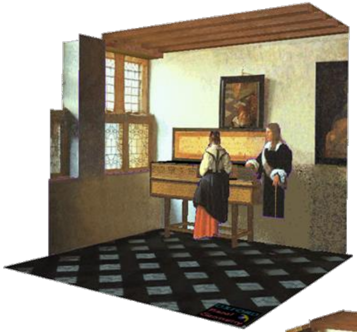
Application: 3D modeling from a single image



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

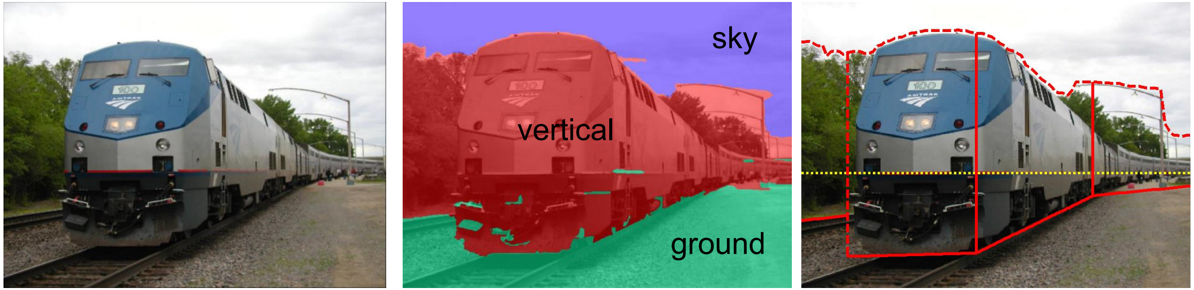
Application: 3D modeling from a single image

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

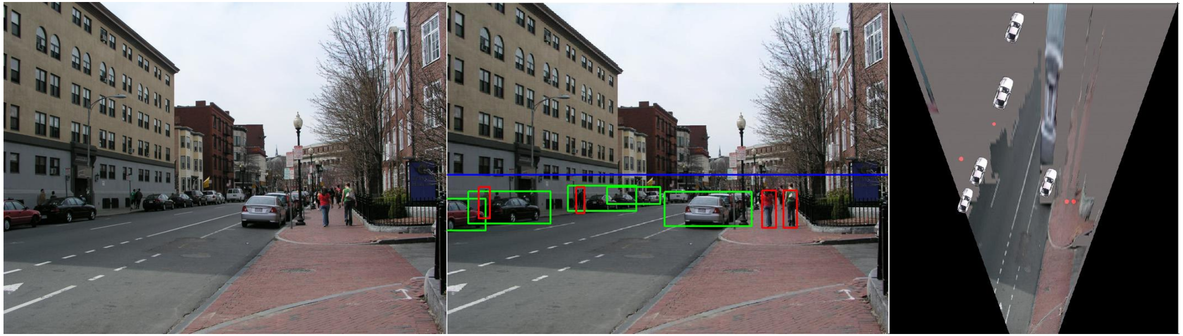
Application: Fully automatic modeling



3D effects

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: Object detection



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

Next Topic

Single Camera Calibration

① 2D - 3D point pairs

② Without

How? → vanishing points

③ Height from an image

- **How about using two cameras?**
  - Prerequisite
    - Review Part2-3: Calibration (this content!)
    - Review Part1-3: Bilateral filtering