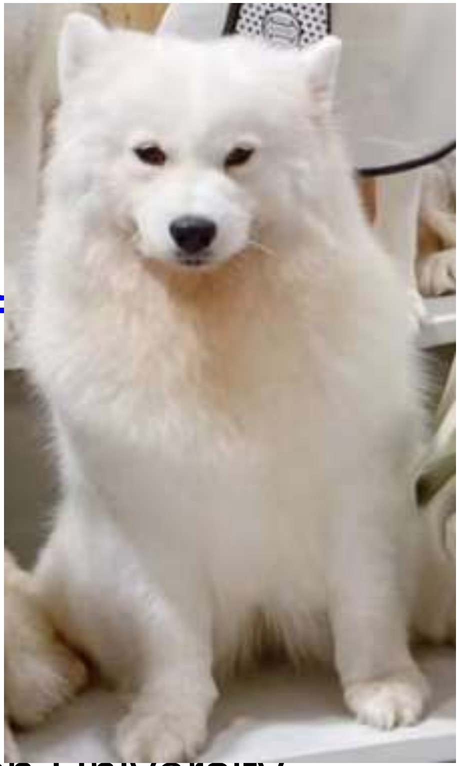
TAs for Computer Vision Ⅰ

2023 - 2024 Fall Semester

王若宇

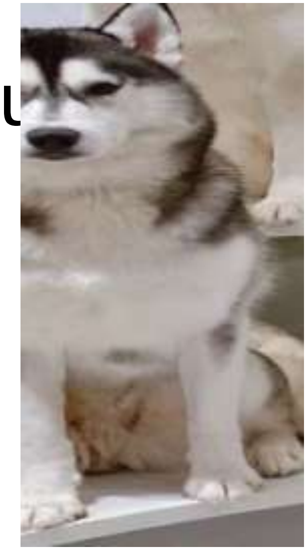
- E-mail: wangry3@shanghaitech.edu.cn
- SIST 1A-306
- Office Hour: Thu. 20:00 - 21:00
- Research Interest:
    - Depth Estimation
- About Me:
    - I am a senior graduated student at ShanghaiTech University.

王晨宇



- E-mail: wangchy8@shanghaitech.edu
- SIST 1A-306
- Office Hour: Tue. 20:00 - 21:00
- Research Interest:
  - Multi-modal Large Language Models
- About Me:
  - I am a senior graduated student at ShanghaiTech University.

赵逸群



- E-mail: zhaoyq12022@shanghaitech.edu
- SIST 1A-306
- Office Hour: Mon. 21:00 - 22:00
- Research Interest:
  - Generative models & Neural Rendering
- About me:
  - I am a second-year graduated student at ShanghaiTech University.

王硕

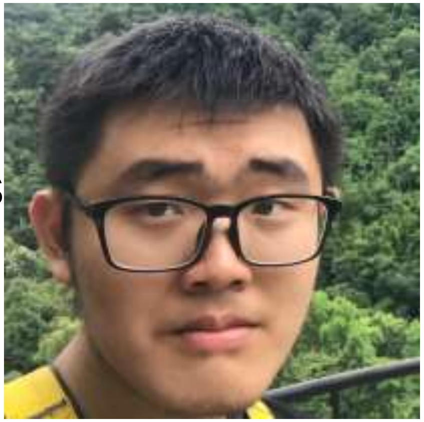
- E-mail: wangshuo2022@shangha
- SIST 1A-306
- Office Hour: Fri. 20:00 - 21:00
- Research Interest:
    - NeRF and Beyond
- About Me:
    - I am a second-year graduated student at S

刘世宇

- E-mail： liushy2023@shanghaitech.
- SIST 1A-306
- Office Hour: Fri. 20:00 - 21:00
- Research Interest:
    - Diffusion Models & Neural Rendering
- About Me:
    - I am a first-year graduated student at ShanghaiTech.

徐京伟

- E-mail: xujw2023@shanghaitech.edu.c
- SIST 1A-306
- Office Hour: Wed. 20:00 - 21:00
- Research Interest:
  - Neural Rendering & Geometry in both Graphics



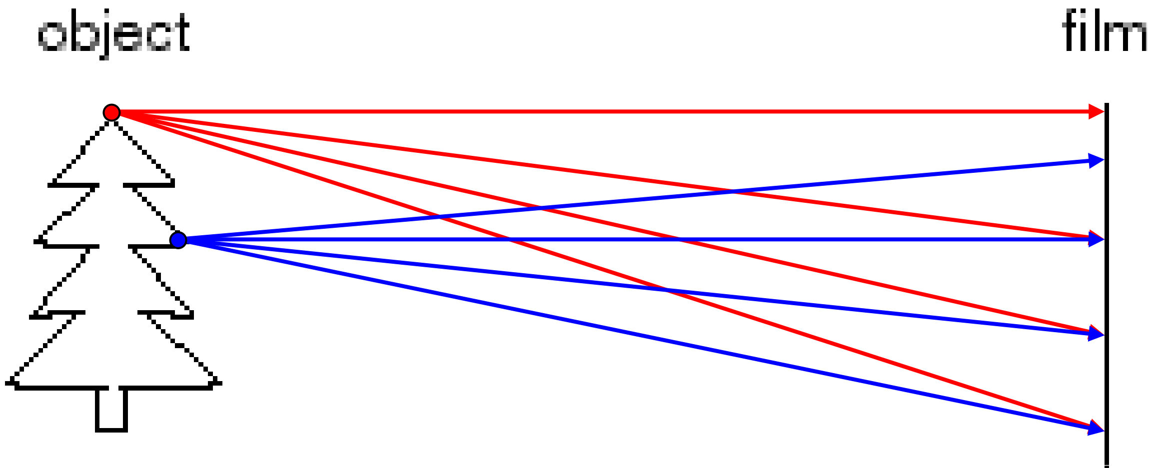
Course Evaluation

- 期末考试（60%）
- 作业：10%
- 随堂测验：10%
- 项目设计：20%

Overview of next two lectures

- The pinhole projection model
  - Qualitative properties
  - Perspective projection matrix
- Cameras with lenses
  - Depth of focus
  - Field of view
  - Lens aberrations
- Digital cameras
  - Sensors
  - Color
  - Artifacts

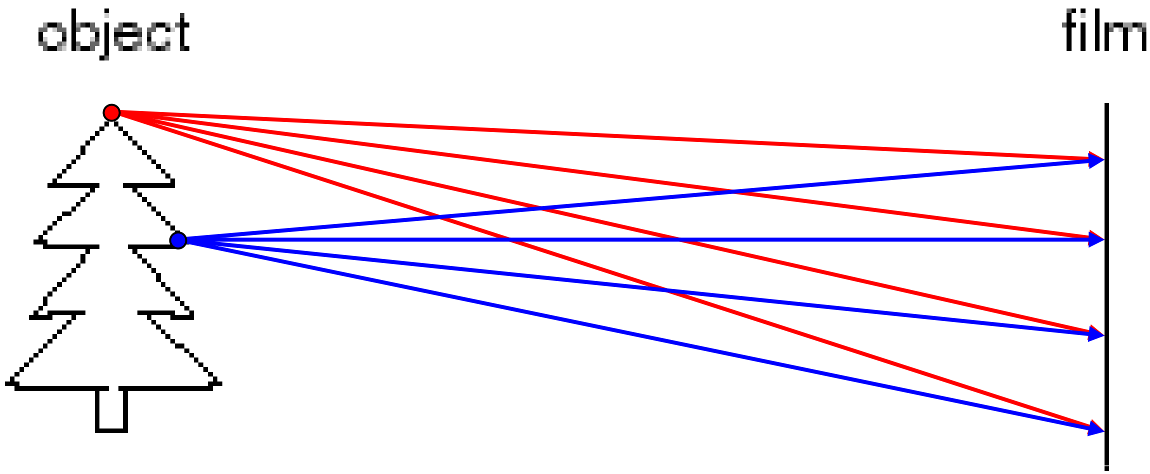
Let's design a camera



Idea 1: put a piece of film in front of an object

Do we get a reasonable image?

Image formation

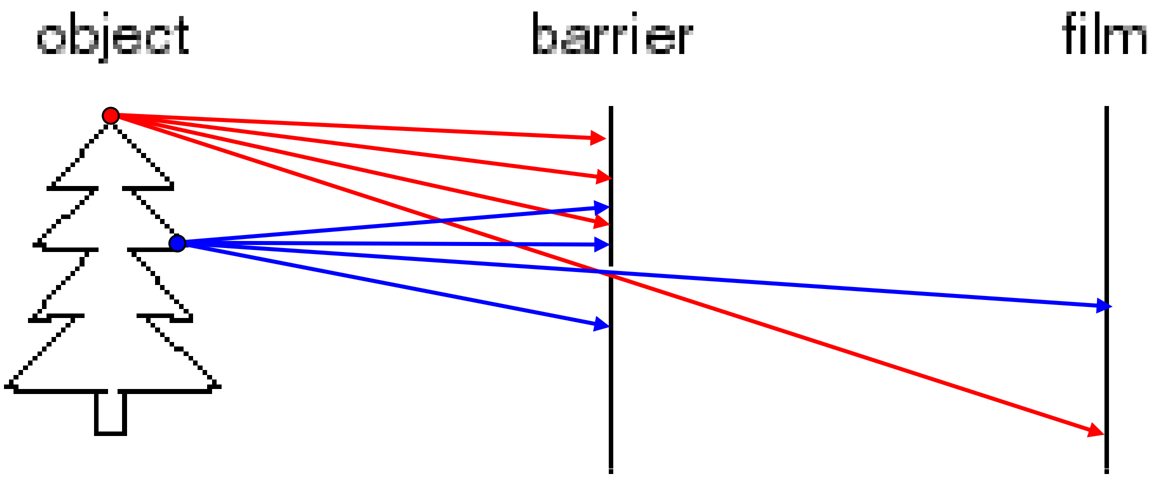


Let’s design a camera

- Idea 1: put a piece of film in front of an object
- Do we get a reasonable image?

Slide source: Seitz

Pinhole camera

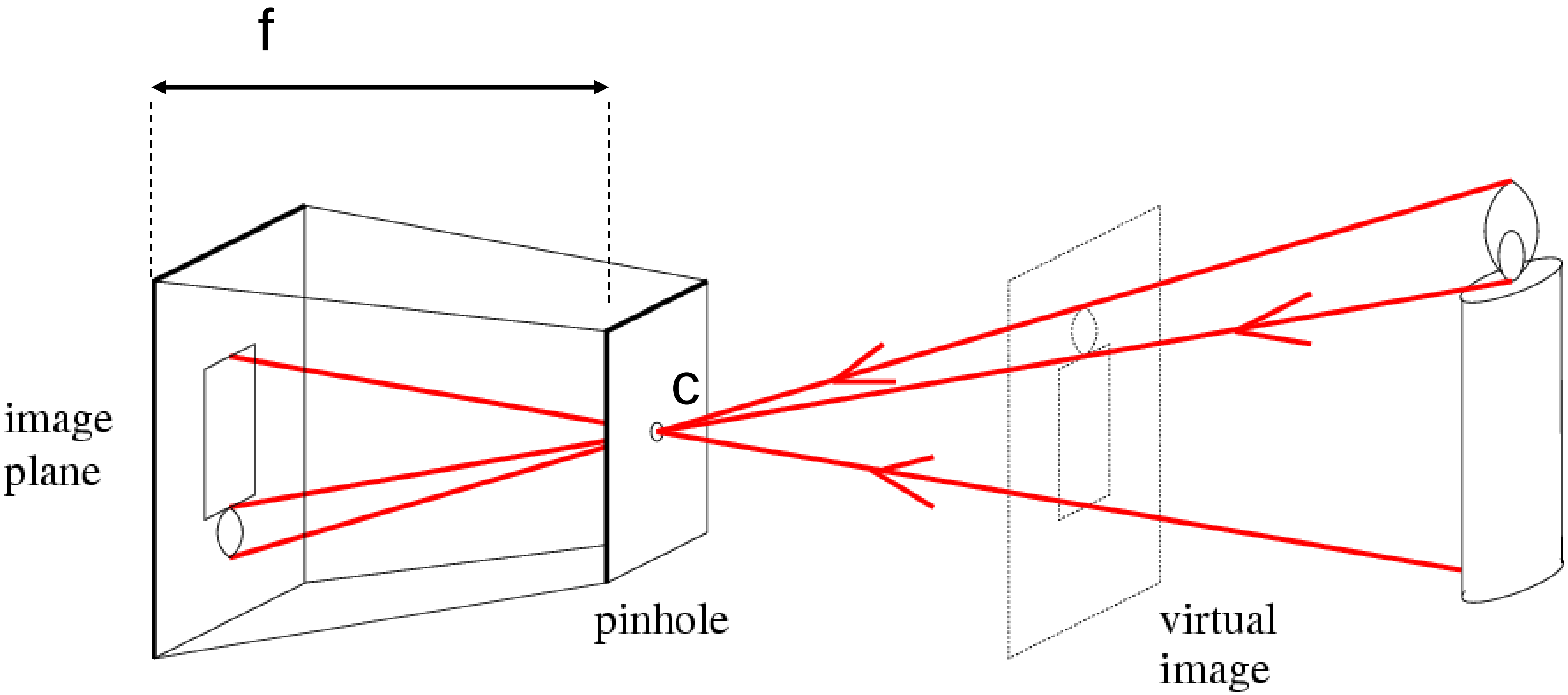


Idea 2: add a barrier to block off most of the rays

- This reduces blurring
- The opening known as the **aperture**

Slide source: Seitz

Pinhole camera

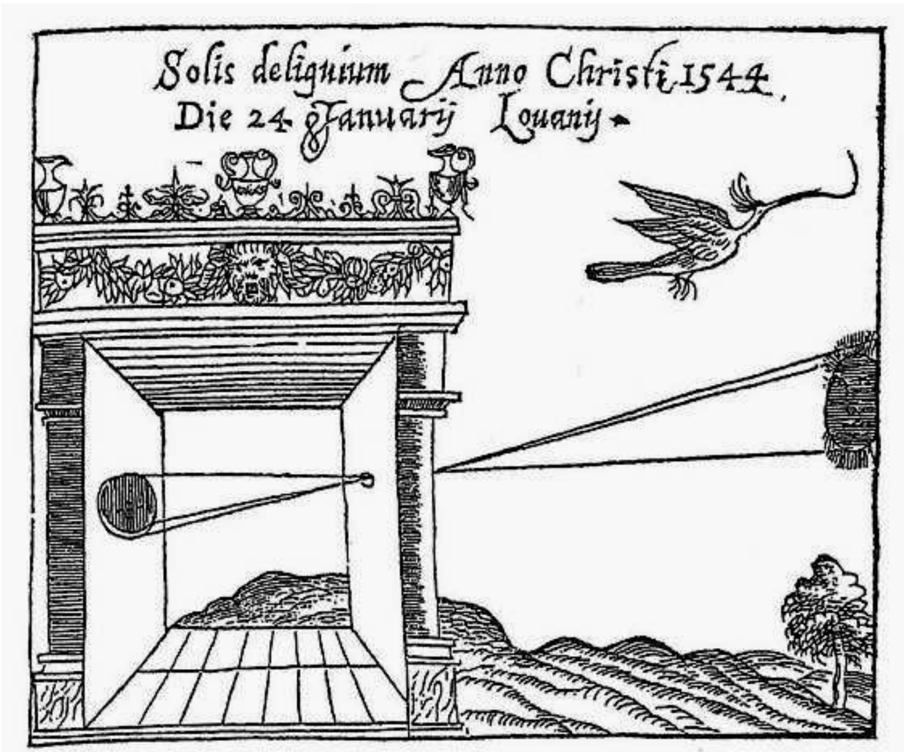


f = focal length

c = center of the camera

Figure from Forsyth

Camera obscura



Gemma Frisius, 1558

- Basic principle known to Mozi (470-390 BCE), Aristotle (384-322 BCE)
- Drawing aid for artists: described by Leonardo da Vinci (1452-1519)

Source: A. Efros

Turning a room into a camera obscura

My hotel room, contrast enhanced.

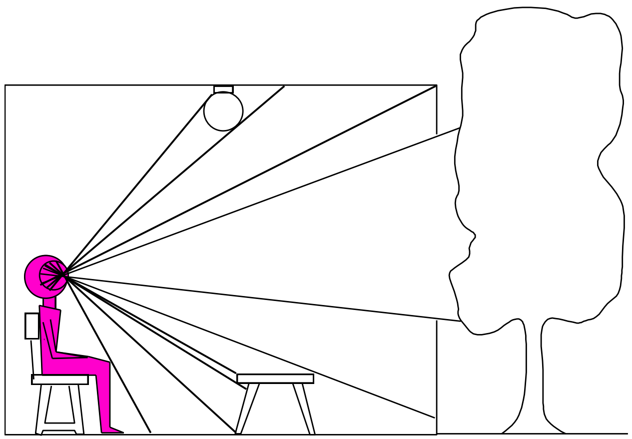
The view from my window

Accidental pinholes produce images that are unnoticed or misinterpreted as shadows

A. Torralba and W. Freeman, [Accidental Pinhole and Pinspeck Cameras](), CVPR 2012

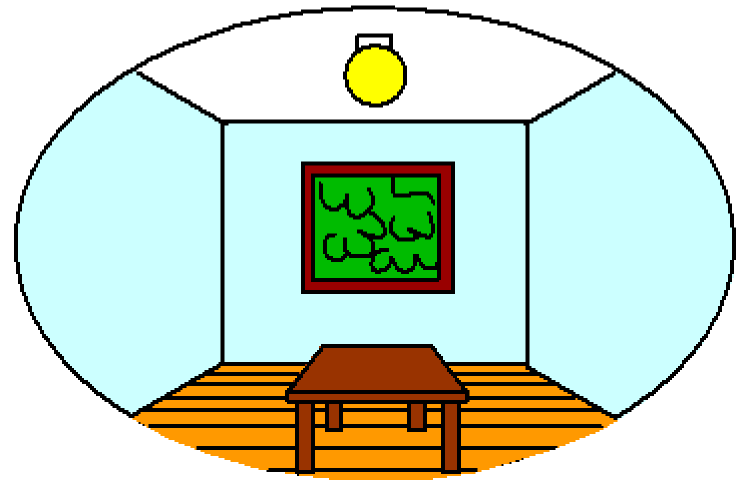
Dimensionality reduction: from 3D to 2D

3D world



Point of observation

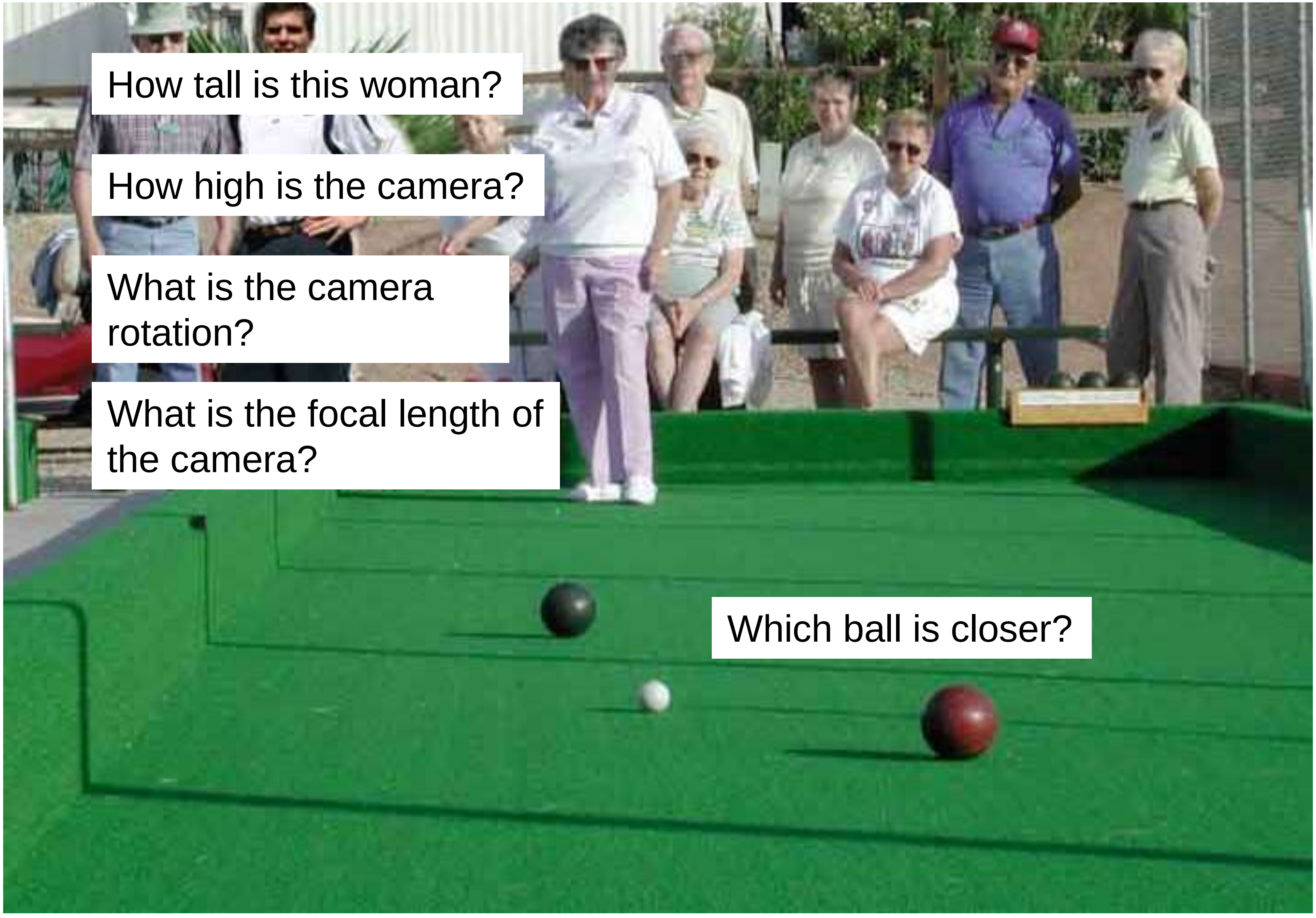
2D image



Slide by A. Efros

Figures © Stephen E. Palmer, 2002

Single-view Geometry



Slide source: Seitz

Projection can be tricky…

Slide source: Seitz

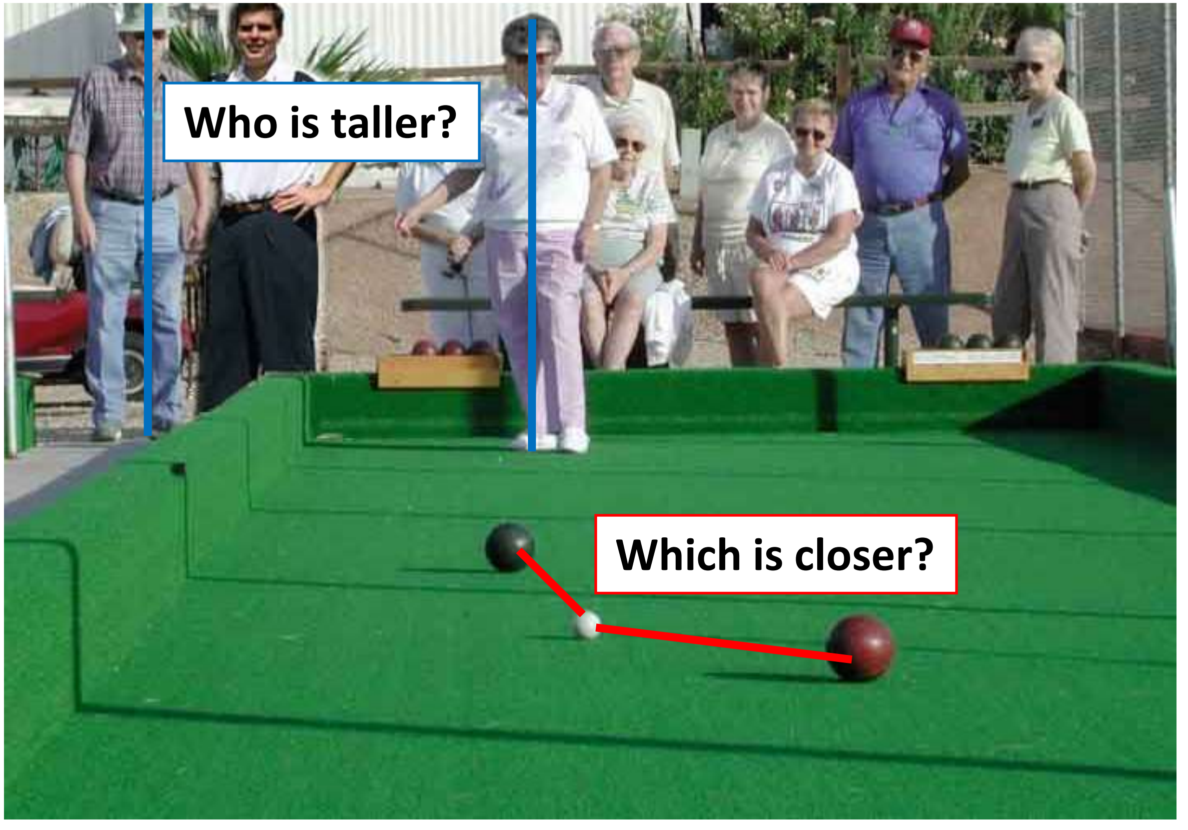
Projection can be tricky…

Making of 3D sidewalk art: http://www.youtube.com/watch?v=3SNYtd0Ayt0

Projective Geometry

What is lost?

- Length



Length is not preserved

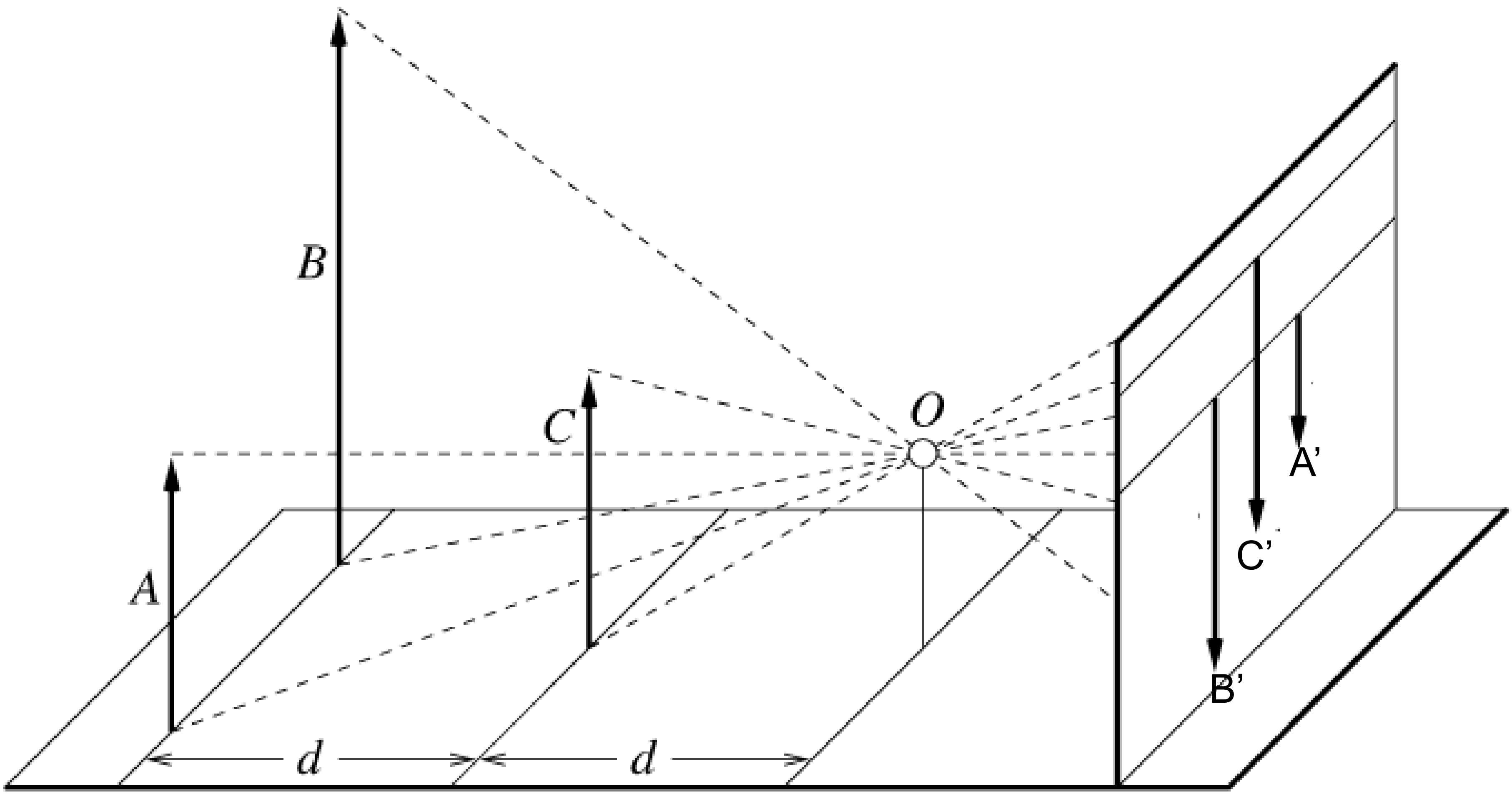
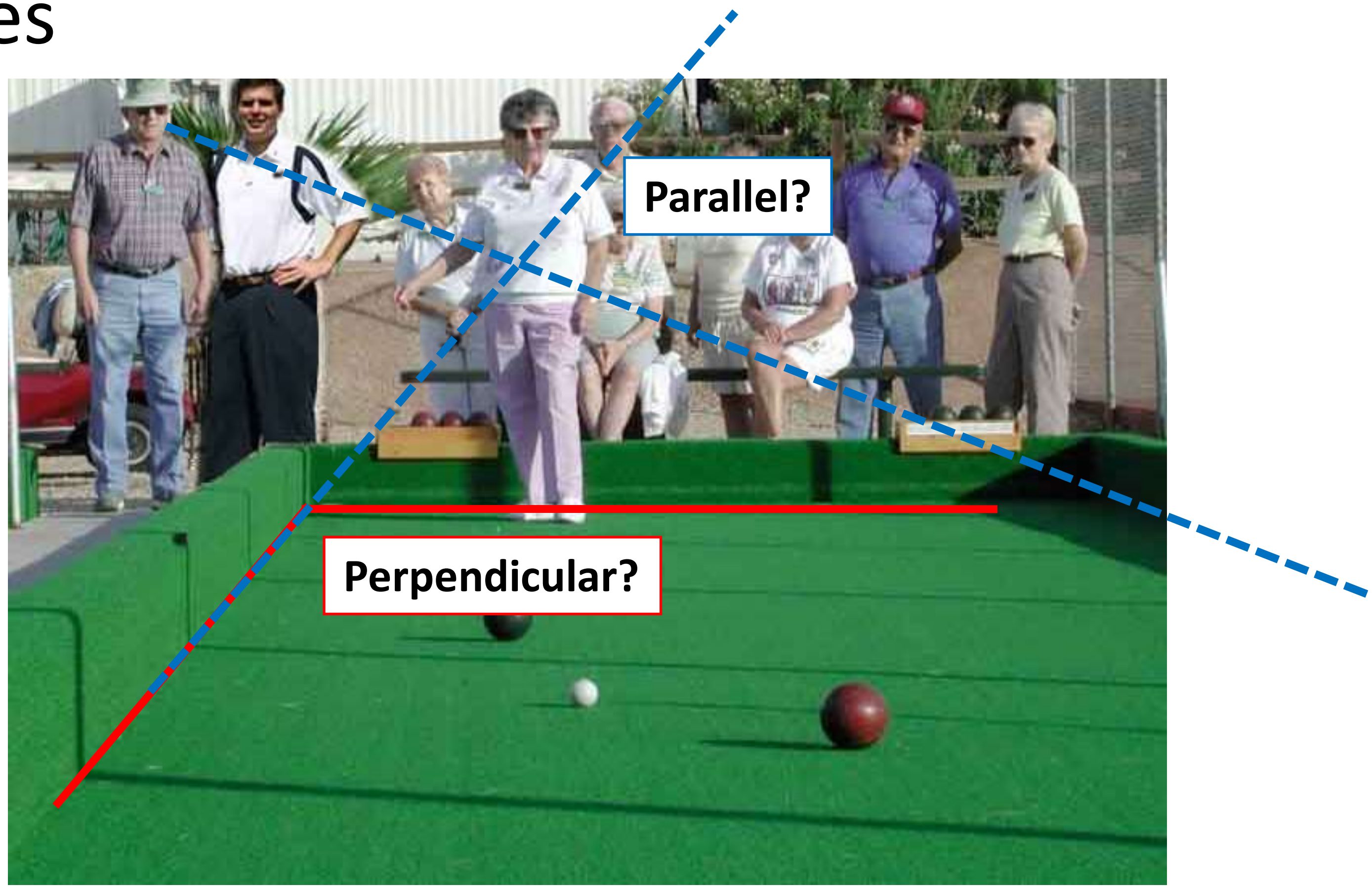


Figure by David Forsyth

Projective Geometry

What is lost?

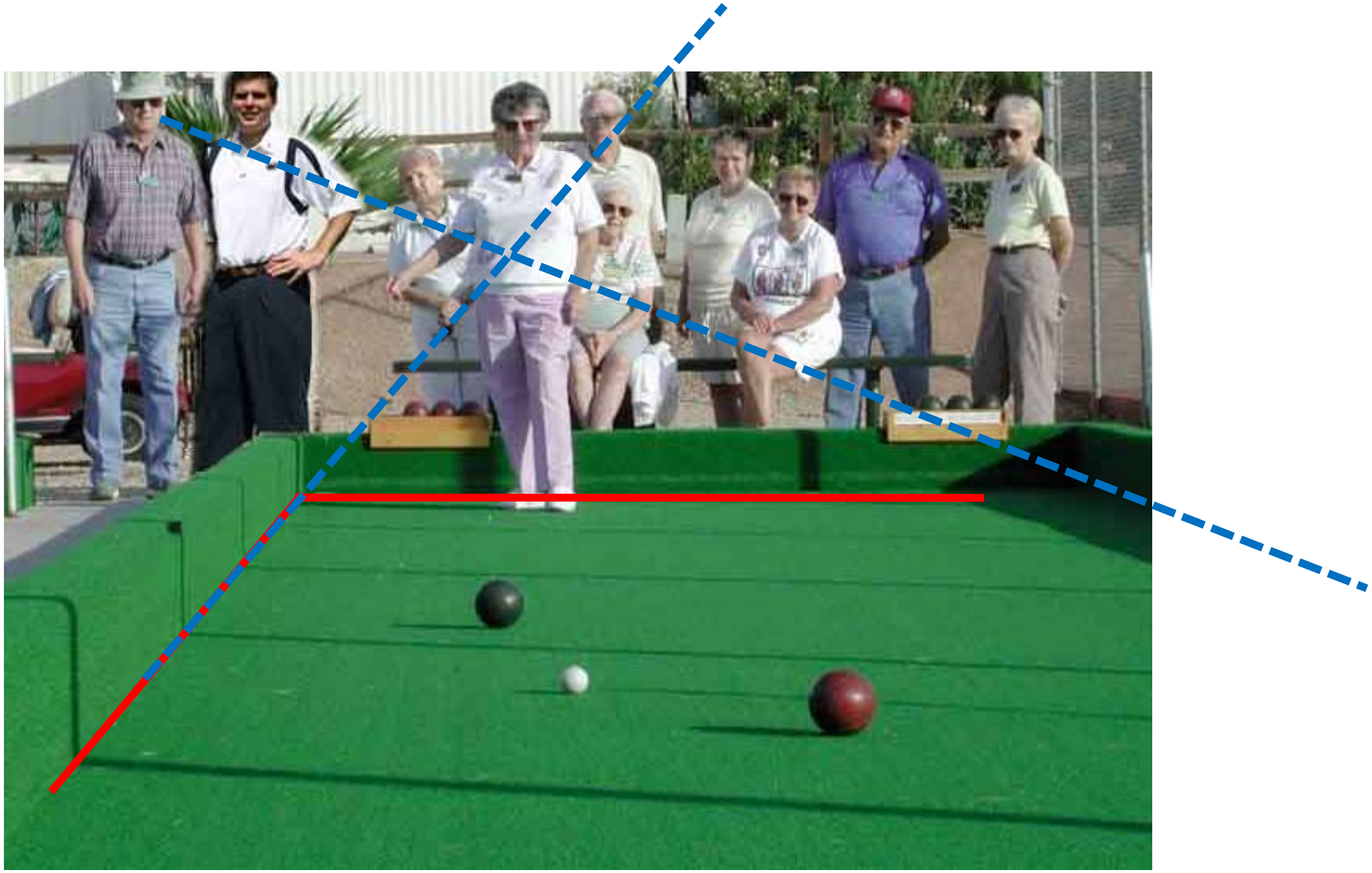
- Length
- Angles



Projective Geometry

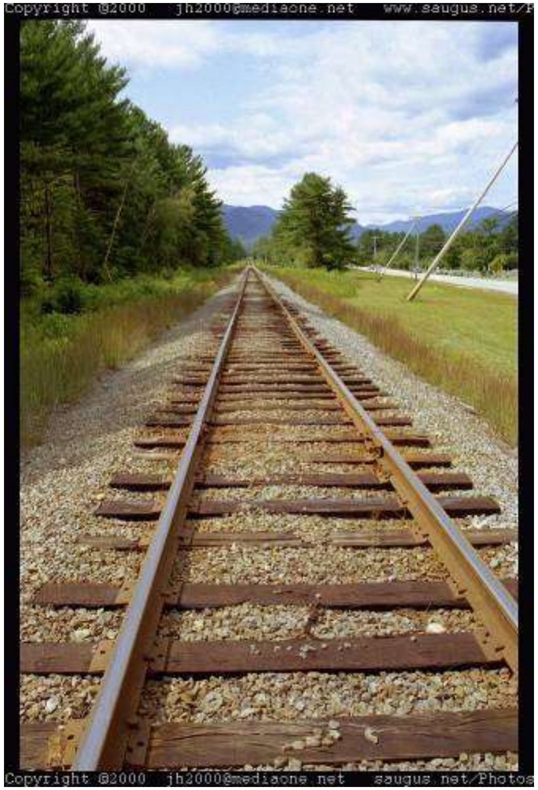
What is preserved?

- Straight lines are still straight

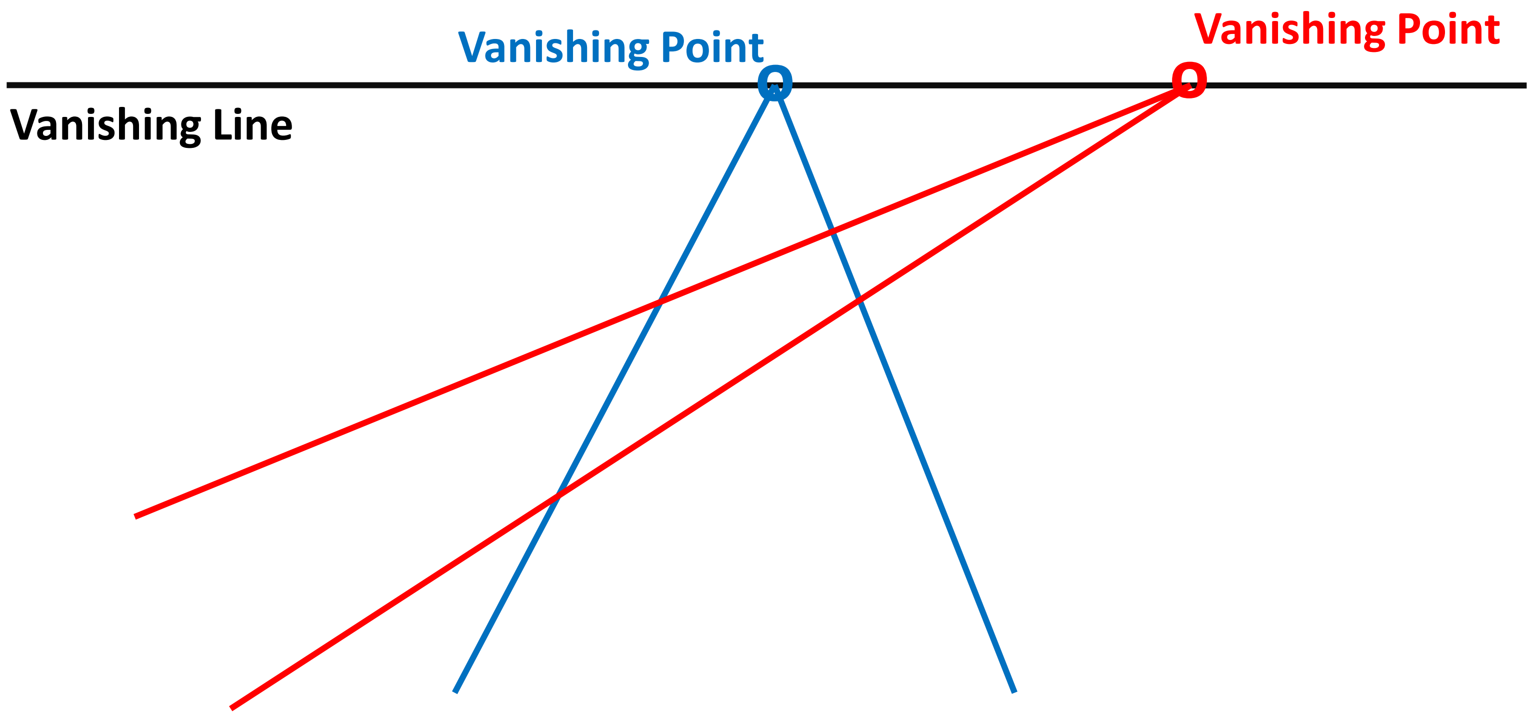


Vanishing points

- All parallel lines converge to a *vanishing point*
  - Each direction in space is associated with its own vanishing point
  - Exception: directions parallel to the image plane

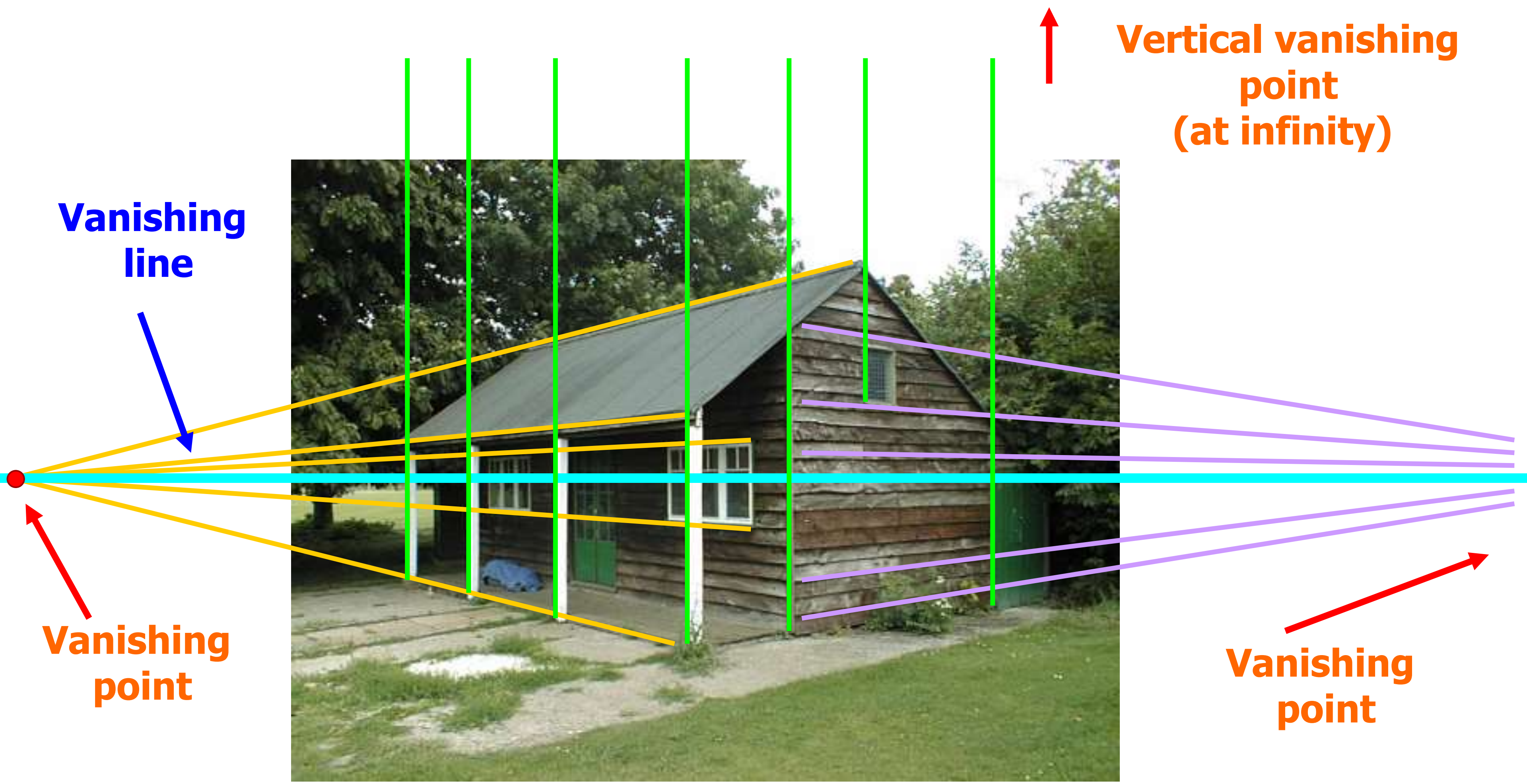


Vanishing points and lines



- The projections of parallel 3D lines intersect at a **vanishing point**
- The projection of parallel 3D planes intersect at a **vanishing line**
- Not all lines that intersect are parallel

Vanishing points and lines



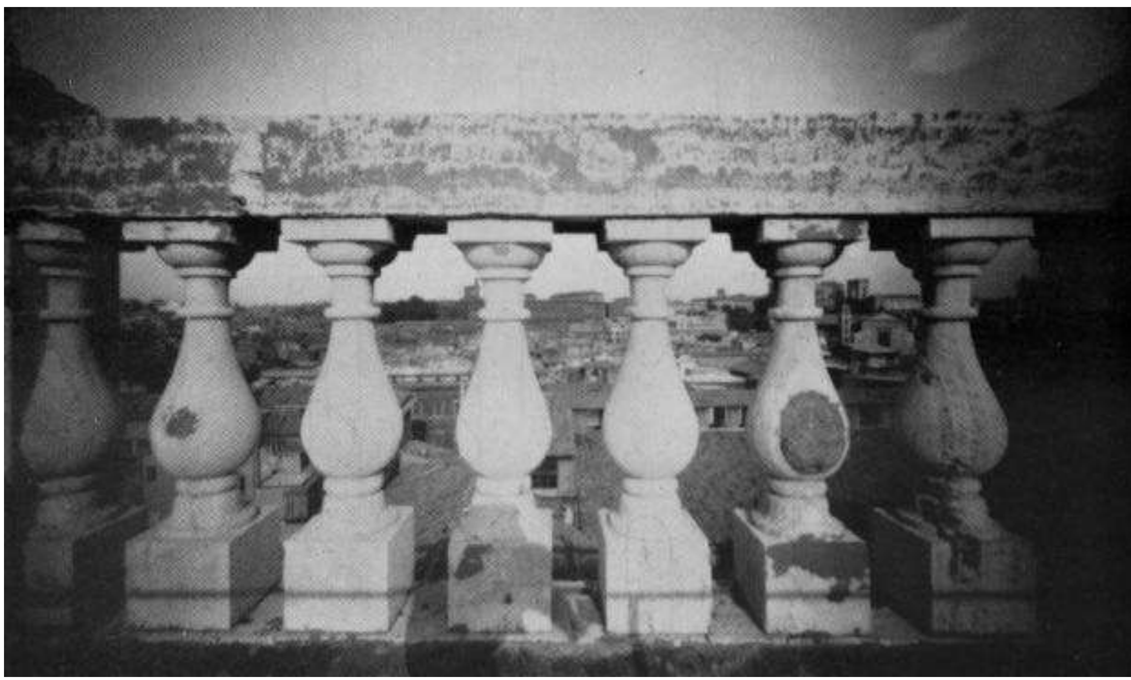
Slide from Efros, Photo from Criminisi

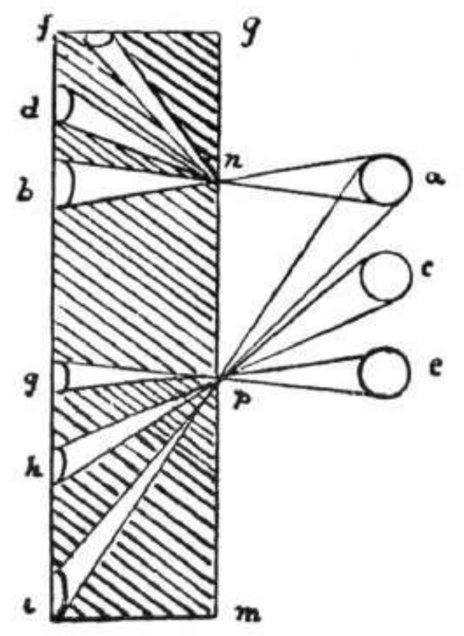
Vanishing objects

Perspective distortion

- Are the widths of the projected columns equal?
  - The exterior columns are wider
  - This is not an optical illusion, and is not due to

a \





Source: F. Durand

Perspective distortion

- What is the shape of the projection of a sphere?

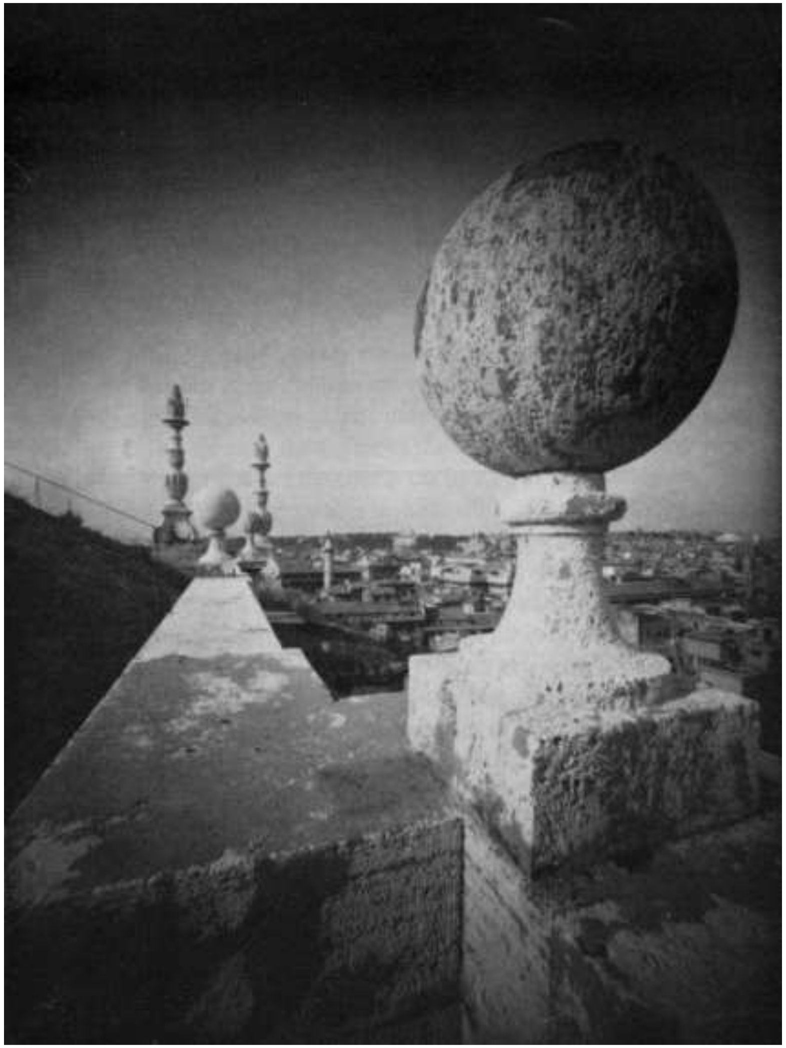
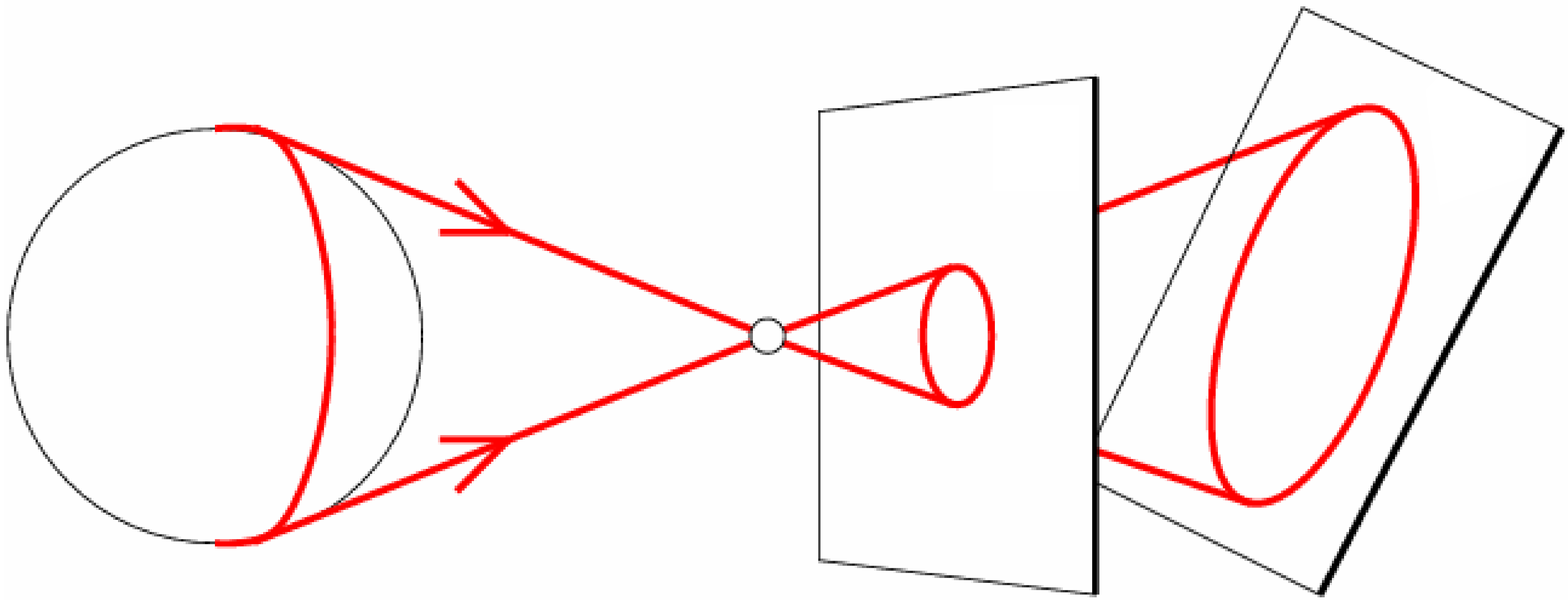


Image source: F. Durand

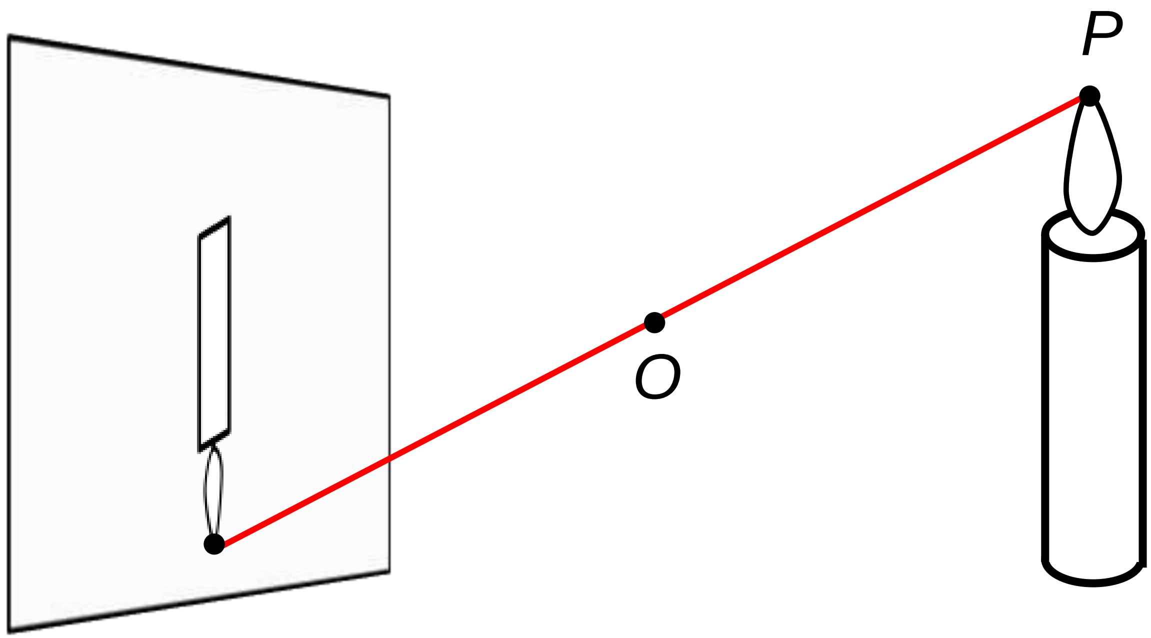
Perspective distortion

- What is the shape of the projection of a sphere?



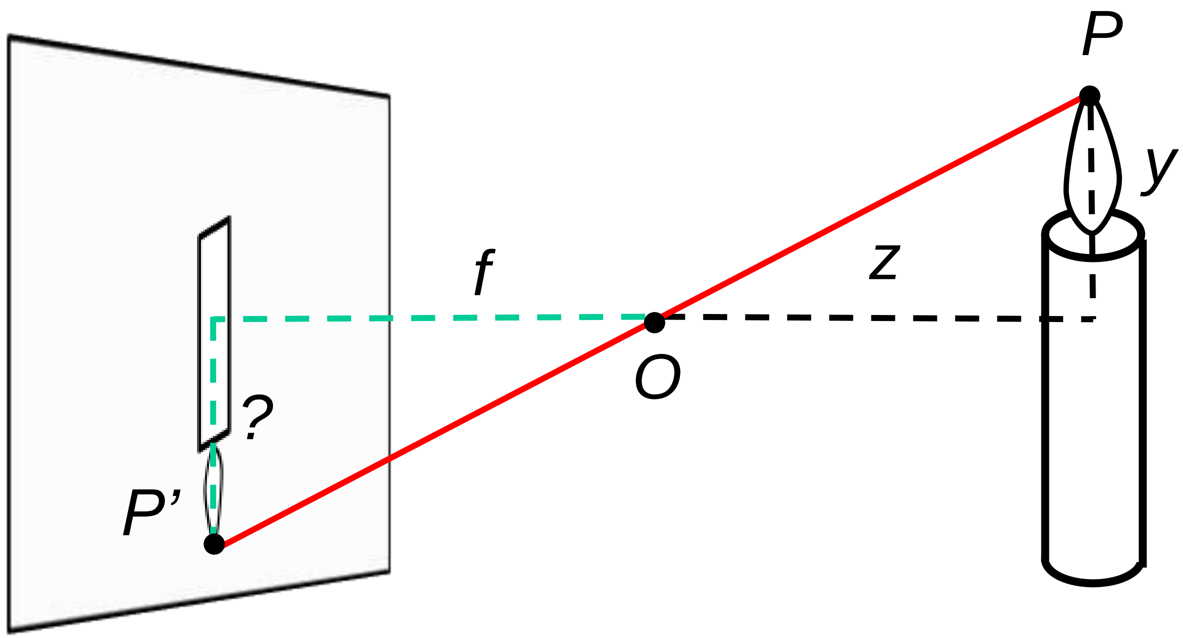
Perspective distortion: People

Modeling projection



- To compute the projection *P'* of a scene point *P*, form the **visual ray** connecting *P* to the camera center *O* and find where it intersects the image plane
  - All scene points that lie on this visual ray have the same projection in the image
  - Are there scene points for which this projection is undefined?

Modeling projection



The coordinate system

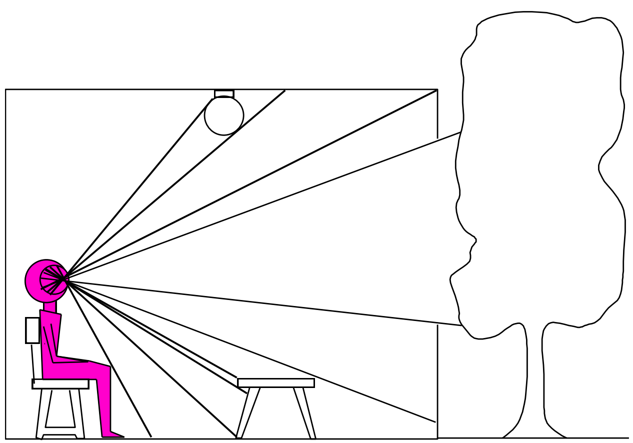
- The optical center (***O***) is at the origin
- The image plane is parallel to xy-plane or perpendicular to the z-axis, which is the *optical axis*

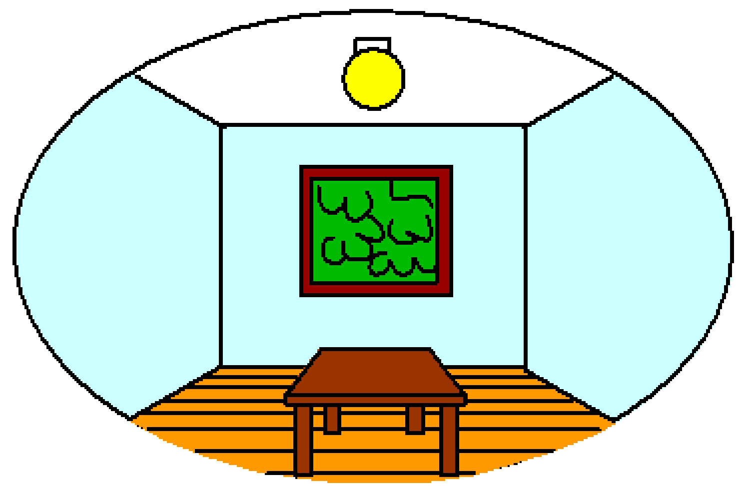
Projection equations

- Derived using similar triangles $(x, y, z) \rightarrow (f\frac{x}{z}, f\frac{y}{z})$

Fronto-parallel planes

- What happens to the projection of a pattern on a plane parallel to the image plane?
  - All points on that plane are at a fixed *depth z*
  - The pattern gets scaled by a factor of *f* / *z*, but angles and ratios of lengths/areas are preserved





$$(x, y, z) \rightarrow (f\,\frac{x}{z}, f\,\frac{y}{z})$$

Fronto-parallel planes

- What happens to the projection of a pattern on a plane parallel to the image plane?
  - All points on that plane are at a fixed *depth z*
  - The pattern gets scaled by a factor of *f* / *z*, but angles and ratios of lengths/areas are preserved

Piero della Francesca, *Flagellation of Christ*, 1455-1460

Jan Vermeer, *The Music Lesson*, 1662-1665

Perspective Projection (pinhole projection)

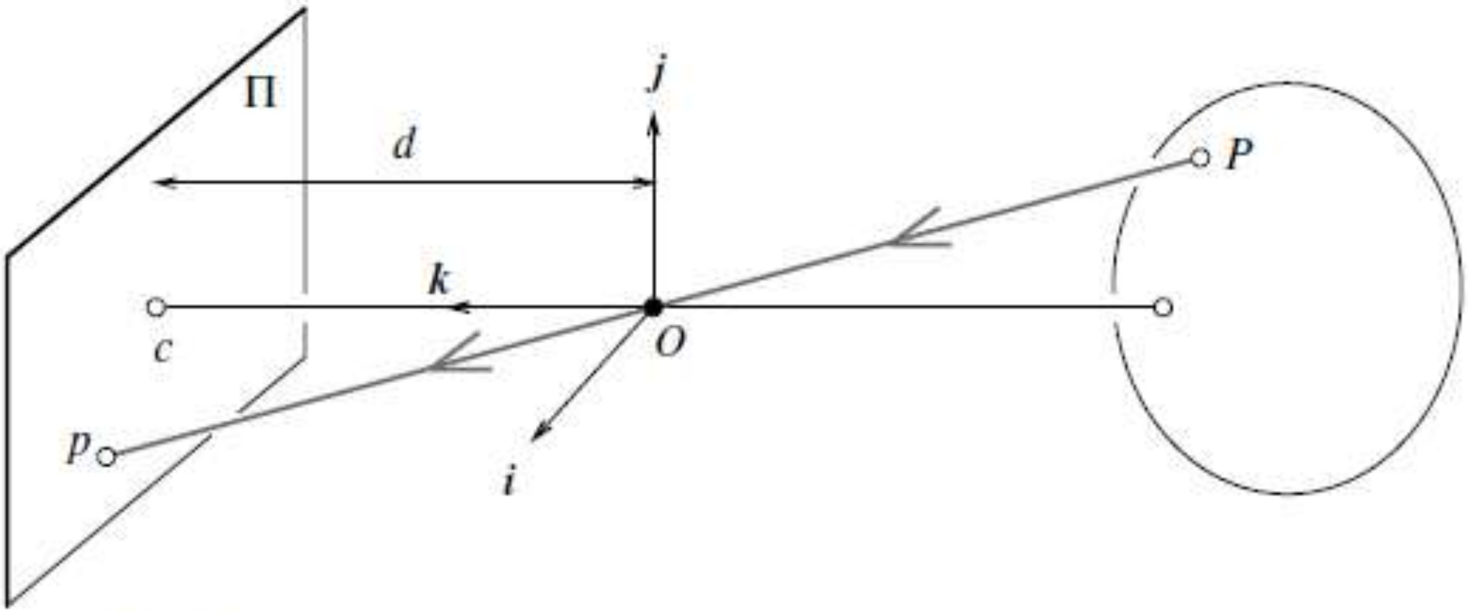


FIGURE 1.4: The perspective projection equations are derived in this section from the collinearity of the point P, its image p, and the pinhole O.

$$\begin{cases} x = \lambda X \\ y = \lambda Y \\ d = \lambda Z \end{cases} \iff \lambda = \frac{x}{X} = \frac{y}{Y} = \frac{d}{Z},$$

$$\begin{cases} x = d\dfrac{X}{Z}, \\ y = d\dfrac{Y}{Z}. \end{cases}$$

Weak Perspective

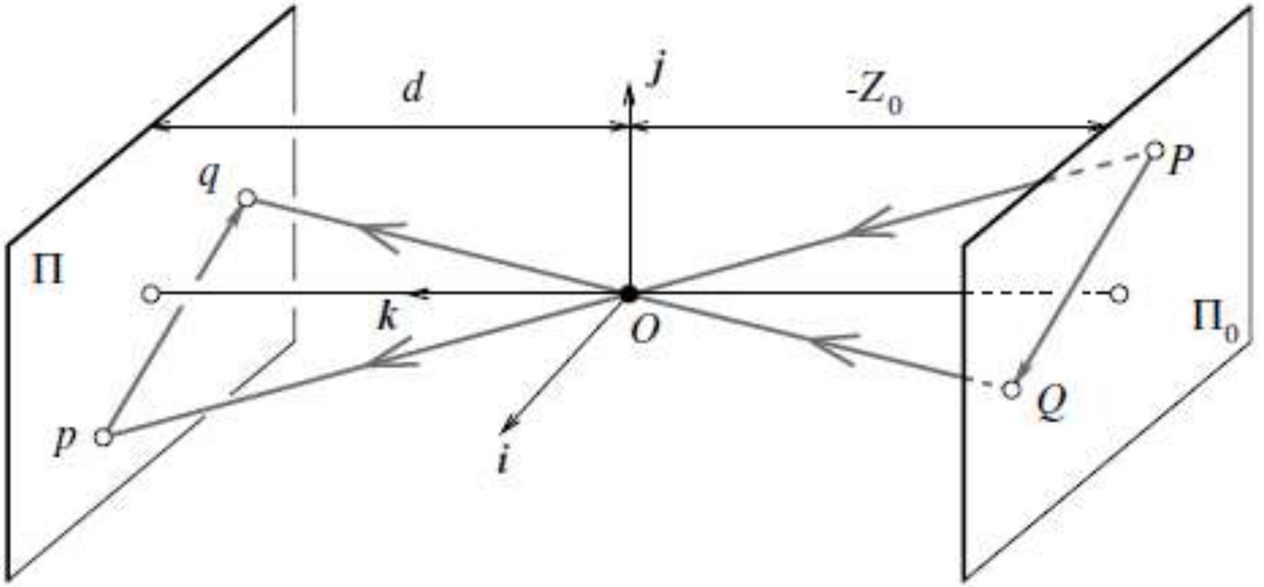


FIGURE 1.5: Weak-perspective projection. All line segments in the plane Π_0 are projected with the same magnification.

$$\begin{cases} x = -mX, \\ y = -mY, \end{cases} \quad \text{where} \quad m = -\frac{d}{Z_0}.$$

Non-homogenous Coordinates

A point P in some coordinate frame (F) = (O, ***i, j, k***) is represented as:

$$\overrightarrow{OP} = X\boldsymbol{i} + Y\boldsymbol{j} + Z\boldsymbol{k}.$$

The same point P in different coordinate systems (A) and (B):

$${}^{A}\boldsymbol{P} = \mathcal{R}\,{}^{B}\boldsymbol{P} + \boldsymbol{t},$$

Here ***R*** is a rotation matrix, ***t*** is a translation vector.

$$\mathcal{R} \stackrel{\text{def}}{=} \left({}^{A}\boldsymbol{i}_B, {}^{A}\boldsymbol{j}_B, {}^{A}\boldsymbol{k}_B\right) = \begin{pmatrix} \boldsymbol{i}_A \cdot \boldsymbol{i}_B & \boldsymbol{j}_A \cdot \boldsymbol{i}_B & \boldsymbol{k}_A \cdot \boldsymbol{i}_B \\ \boldsymbol{i}_A \cdot \boldsymbol{j}_B & \boldsymbol{j}_A \cdot \boldsymbol{j}_B & \boldsymbol{k}_A \cdot \boldsymbol{j}_B \\ \boldsymbol{i}_A \cdot \boldsymbol{k}_B & \boldsymbol{j}_A \cdot \boldsymbol{k}_B & \boldsymbol{k}_A \cdot \boldsymbol{k}_B \end{pmatrix}$$

Homogenous Coordinates

Add one more coordinate:

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

homogeneous image coordinates

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

homogeneous scene coordinates

By introducing homogenous coordinates, we have

$$^{A}\boldsymbol{P} = \mathcal{T}\,^{B}\boldsymbol{P}, \quad \text{where} \quad \mathcal{T} = \begin{pmatrix} \mathcal{R} & \boldsymbol{t} \\ \boldsymbol{0}^{T} & 1 \end{pmatrix},$$

Projection Equation in Homogenous Coordinates

For a point **P** in some fixed world coordinate

P=(X, Y, Z, 1)^T, and its image **p** in the camera's reference frame (normalized image plane) \hat{*p*}= (x,y,1)^T, the projection equation is represented as:

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

Intrinsic Parameters

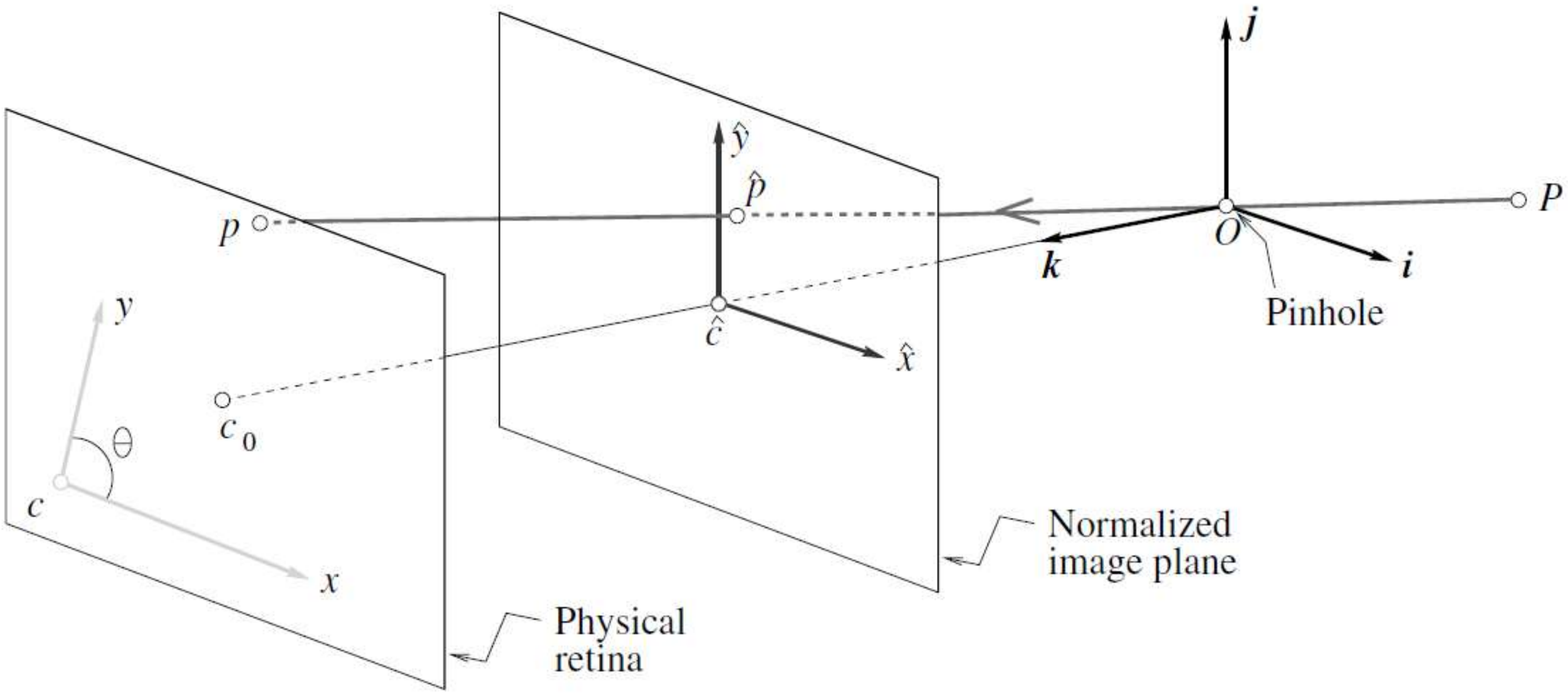
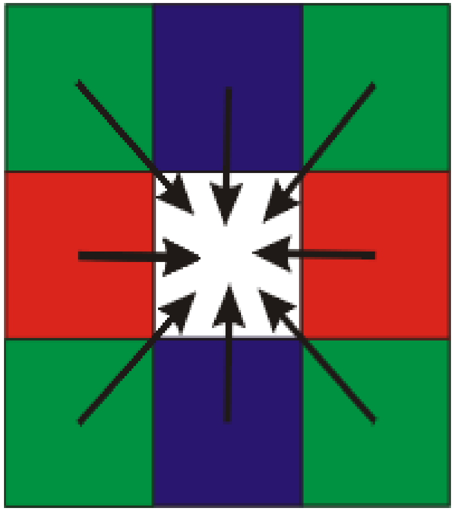


FIGURE 1.14: Physical and normalized image coordinate systems.

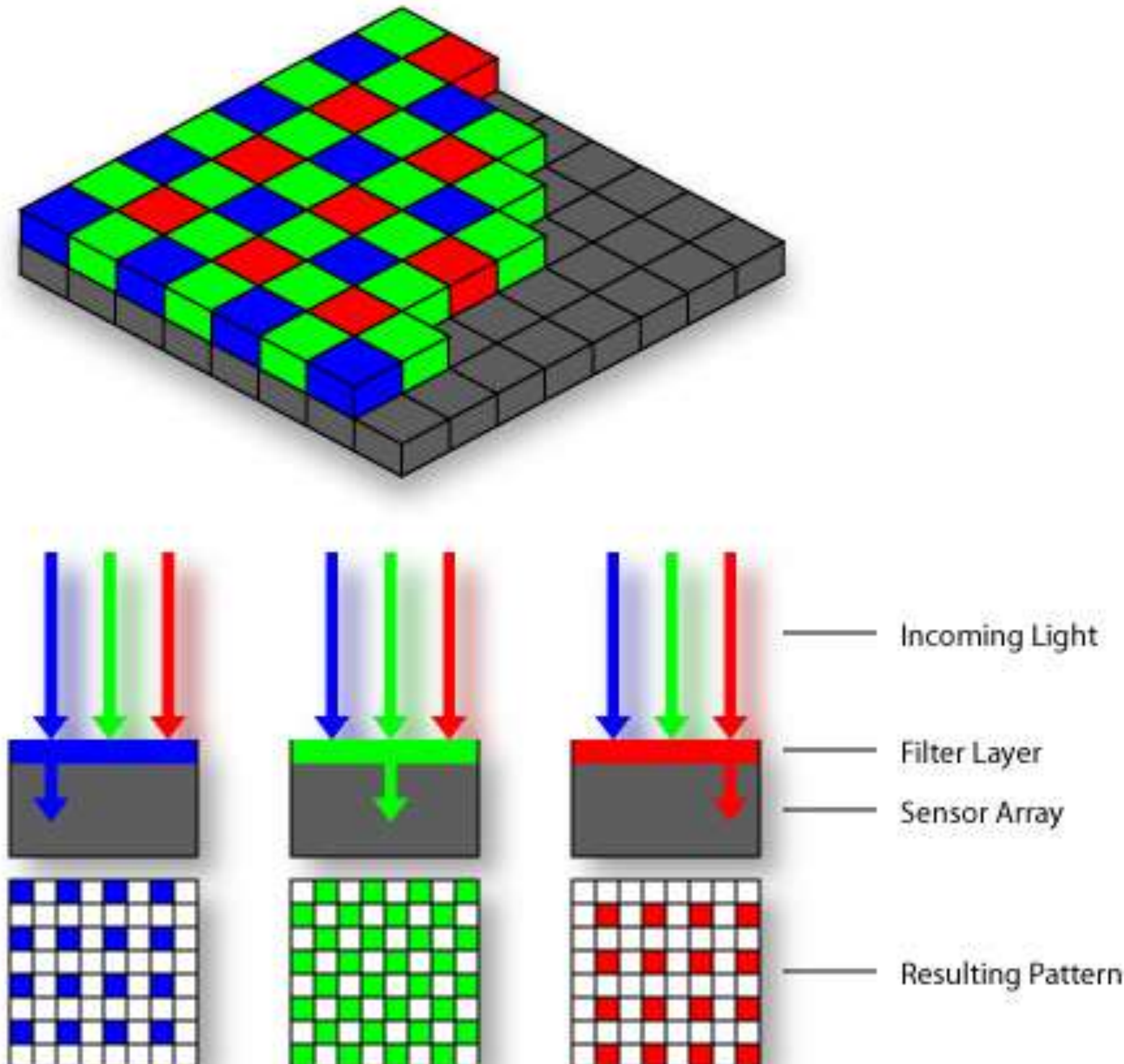
Point at normalized image plan

$$\begin{cases} \hat{x} = \dfrac{X}{Z} \\ \hat{y} = \dfrac{Y}{Z} \end{cases} \Longleftrightarrow \hat{\boldsymbol{p}} = \frac{1}{Z}\begin{pmatrix}\mathrm{Id} & \mathbf{0}\end{pmatrix}\boldsymbol{P}$$

Color Sensing: Bayer Grid



Estimate RGB at each cell from neighboring values



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

Slide by Steve Seitz

Intrinsic Parameters

- The coordinates (x, y) of the image point p are expressed in pixel units (not meters).
- Pixels may be rectangular instead of square(skewed).

$$\begin{cases} x = kf\dfrac{X}{Z} = kf\hat{x}, \\ y = lf\dfrac{Y}{Z} = lf\hat{y}. \end{cases} \qquad \alpha = kf \text{ and } \beta = lf$$

- The center of the CCD matrix usually does not coincide with the image center c_0

$$\begin{cases} x = \alpha\hat{x} + x_0, \\ y = \beta\hat{y} + y_0. \end{cases}$$

- Due to manufacturing error, the angle between two image axes is not 90 degrees.

$$\begin{cases} x = \alpha\hat{x} - \alpha\cot\theta\hat{y} + x_0, \\ y = \dfrac{\beta}{\sin\theta}\hat{y} + y_0. \end{cases}$$

Intrinsic Parameters

Putting all equations together, we get

$$\boldsymbol{p} = \mathcal{K}\hat{\boldsymbol{p}}, \quad \text{where} \quad \boldsymbol{p} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad \text{and} \quad \mathcal{K} \stackrel{\text{def}}{=} \begin{pmatrix} \alpha & -\alpha \cot\theta & x_0 \\ 0 & \dfrac{\beta}{\sin\theta} & y_0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Here $\mathcal{K}$ is called (Internal) calibration matrix of the camera.

$$\boldsymbol{p} = \frac{1}{Z}\mathcal{K}(\mathrm{Id} \quad \mathbf{0})\boldsymbol{P} = \frac{1}{Z}\mathcal{M}\boldsymbol{P}, \quad \text{where} \quad \mathcal{M} \stackrel{\text{def}}{=} (\mathcal{K} \quad \mathbf{0}).$$

Intrinsic parameters: $\alpha,\ \beta,\ \theta,\ x_0,$ and y_0

Extrinsic Parameters:

Camera coordinate frame:C

$$p = \frac{1}{Z}\mathcal{M}^C P$$

World coordinate frame:W

$${}^C\boldsymbol{P} = \begin{pmatrix} \mathcal{R} & \boldsymbol{t} \\ \boldsymbol{0}^T & 1 \end{pmatrix} {}^W\boldsymbol{P},$$

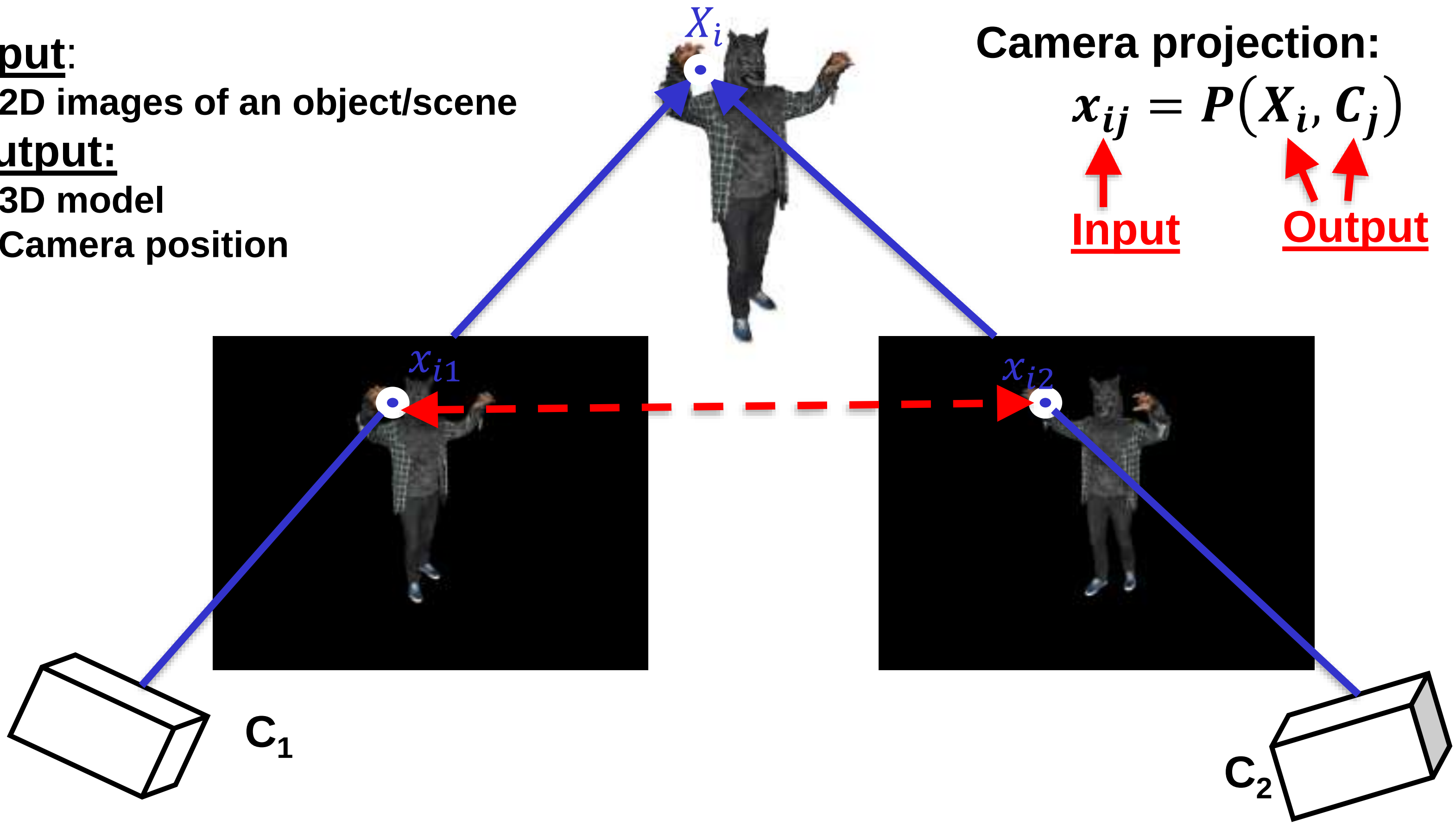
Taking P = ^{W}P

$$\boldsymbol{p} = \frac{1}{Z}\mathcal{M}\boldsymbol{P}, \quad \text{where} \quad \mathcal{M} = \mathcal{K}\begin{pmatrix}\mathcal{R} & \boldsymbol{t}\end{pmatrix}.$$

A revisit of 3D reconstruction

<u>Input</u>:
- **2D images of an object/scene**

<u>Output:</u>
- **3D model**
- **Camera position**



Camera projection:

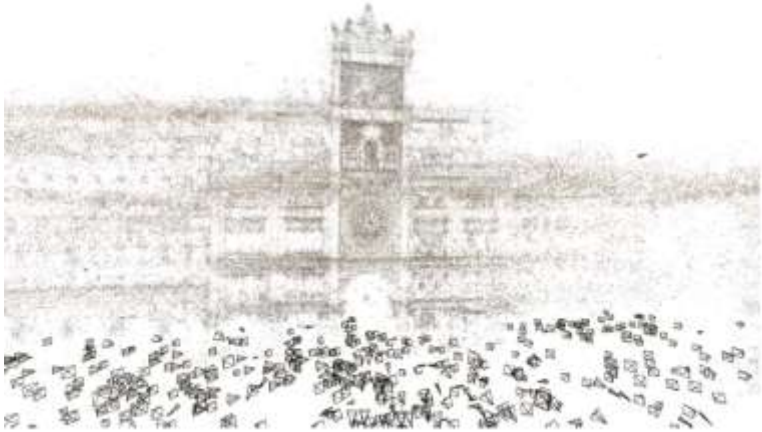
$$x_{ij} = P(X_i, C_j)$$

<u>Input</u> **<u>Output</u>**

A Revisit of 3D Reconstruction

- Correct point matching is important.
  - 3D reconstruction is an optimization problem.
  - Objective function of reprojection.

$$\min_{X_i\,C_j} \sum_{ij} ||X_{ij} - P(\mathrm{X}_i, C_j)||$$



San Marco Square, 14,079 images, 4,515,157 points

Building Rome in a Day. Sameer Agarwal, Noah Snavely, Ian Simon, Steven M. Seitz and Richard Szeliski. Communications of the ACM, 2011.

Extrinsic Parameters

Extrinsic Parameters: 3 independent parameters in rotation matrix ***R*** and 3 parameters in translation vector ***t.***

Denote the columns in M as $\boldsymbol{m}_1^T$, $\boldsymbol{m}_2^T$ and $\boldsymbol{m}_3^T$

Then we have

$$\begin{cases} x = \dfrac{\boldsymbol{m}_1 \cdot \boldsymbol{P}}{\boldsymbol{m}_3 \cdot \boldsymbol{P}}, \\ y = \dfrac{\boldsymbol{m}_2 \cdot \boldsymbol{P}}{\boldsymbol{m}_3 \cdot \boldsymbol{P}}. \end{cases}$$

the three rows $\boldsymbol{r}_1^T$, $\boldsymbol{r}_2^T$, and $\boldsymbol{r}_3^T$ of the matrix $\mathcal{R}$,
three coordinates t_1, t_2, and t_3 of the vector $\boldsymbol{t}$,

$$\mathcal{M} = \begin{pmatrix} \alpha \boldsymbol{r}_1^T - \alpha \cot\theta \boldsymbol{r}_2^T + x_0 \boldsymbol{r}_3^T & \alpha t_1 - \alpha \cot\theta t_2 + x_0 t_3 \\ \dfrac{\beta}{\sin\theta} \boldsymbol{r}_2^T + y_0 \boldsymbol{r}_3^T & \dfrac{\beta}{\sin\theta} t_2 + y_0 t_3 \\ \boldsymbol{r}_3^T & t_3 \end{pmatrix}.$$

Projection matrix

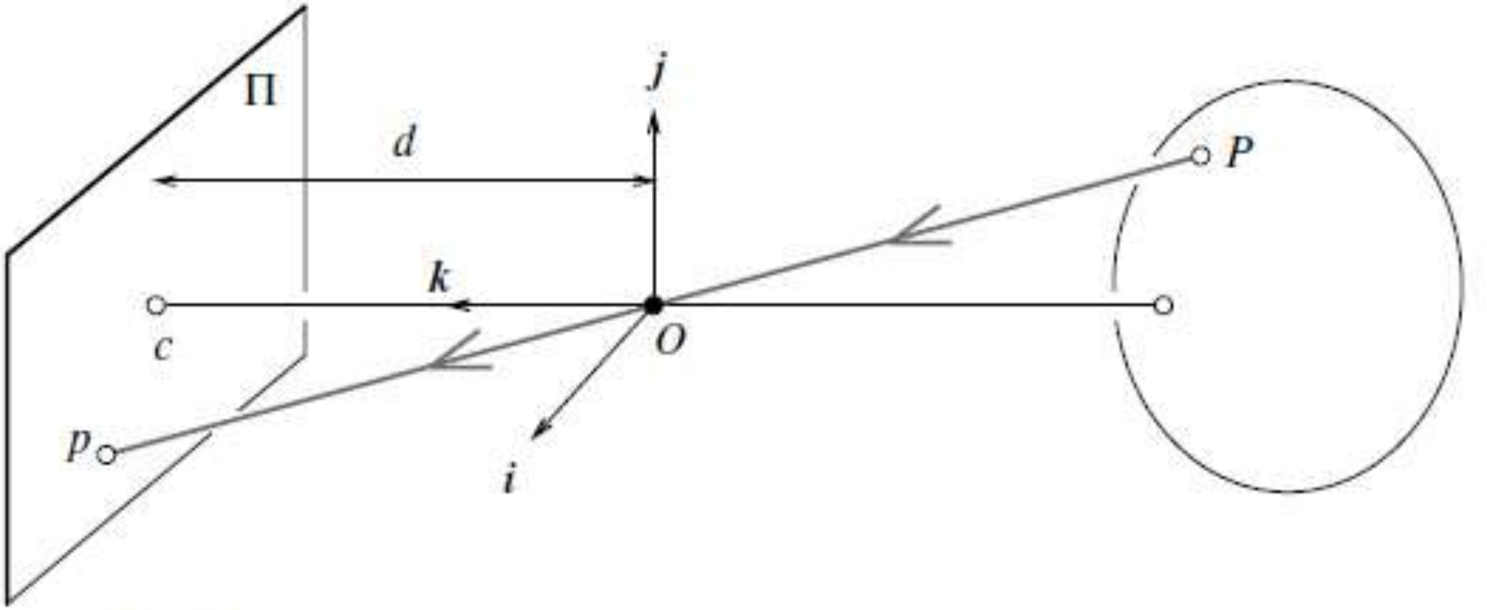


FIGURE 1.4: The perspective projection equations are derived in this section from the collinearity of the point P, its image p, and the pinhole O.

- Unit aspect ratio
- Optical center at (0,0)
- No skew
- No rotation
- Camera at (0,0,0)

$$\boldsymbol{p} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{P} \Rightarrow \begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}\underbrace{\begin{matrix}f & 0 & 0\\0 & f & 0\\0 & 0 & 1\end{matrix}}_{\mathbf{K}} & \begin{matrix}0\\0\\0\end{matrix}\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

Slide Credit: Saverese

Remove assumption: known optical center

Intrinsic Assumptions
- Unit aspect ratio
- No skew

Extrinsic Assumptions
- No rotation
- Camera at (0,0,0)

$$\boldsymbol{p} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{P} \Rightarrow \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Remove assumption: square pixels

Intrinsic Assumptions
- No skew

Extrinsic Assumptions
- No rotation
- Camera at (0,0,0)

$$\boldsymbol{p} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{P} \Rightarrow w\begin{bmatrix}u \\ v \\ 1\end{bmatrix} = \begin{bmatrix}\alpha & 0 & u_0 & 0 \\ 0 & \beta & v_0 & 0 \\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}x \\ y \\ z \\ 1\end{bmatrix}$$

Remove assumption: non-skewed pixels

Intrinsic Assumptions

Extrinsic Assumptions
- No rotation
- Camera at (0,0,0)

$$\boldsymbol{p} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{P} \Rightarrow \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 & 0 \\ 0 & \beta & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Note: different books use different notation for parameters

Degrees of freedom

$$\boldsymbol{p} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{P}$$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \overset{5}{\begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}} \overset{6}{\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$