Image Formation - Geometry

Geometry Aspect

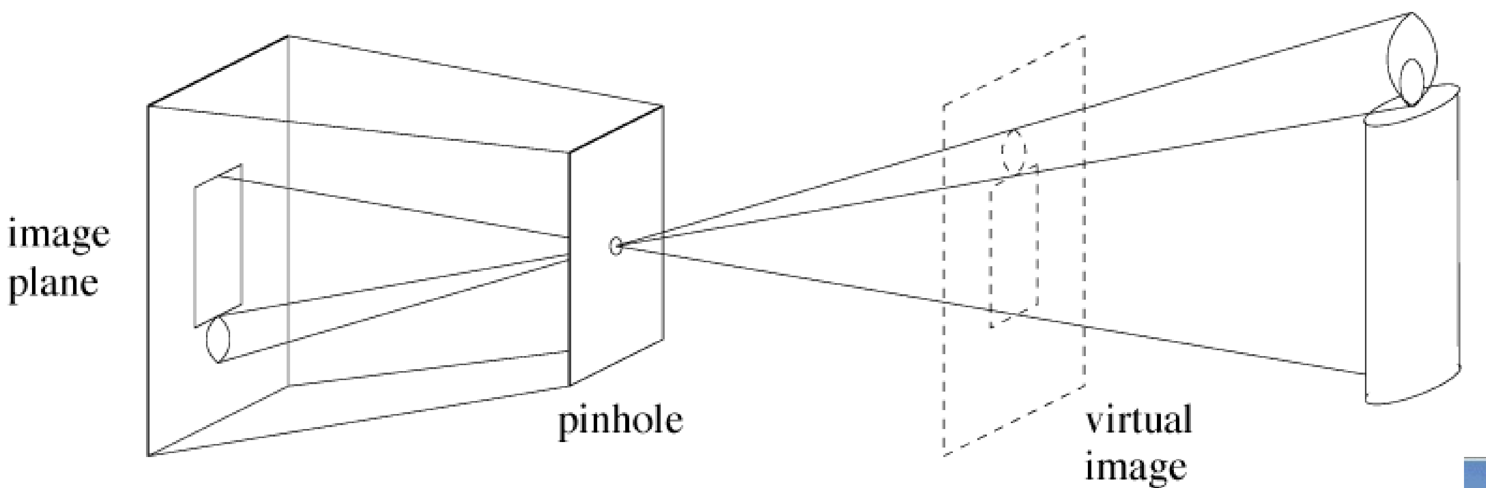
- Where does the image of a point end up with?
- Geometry transform
  - A unique mapping
- Lens effect
  - Both geometrical and color distortion
  - Not necessarily a unique mapping

Camera geometry

- Projection models
  - *Where* are points in the world imaged? That is, where do they show up in the image?
- The point P = (x, y, z) shows up at P′ = (x', y')
  - What is the relationship between P and P' ?
  - P′ = f_{proj}(P)
    - What is f_{proj}? Is it linear, nonlinear…?
- Different projection models determine different f_{proj}*'s*
  - Why is there more than one? Is one "right"?
  - What does a real camera do?

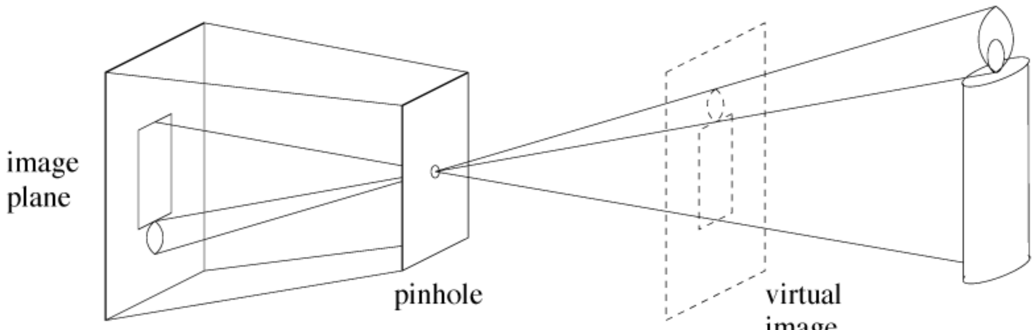
Pinhole camera model

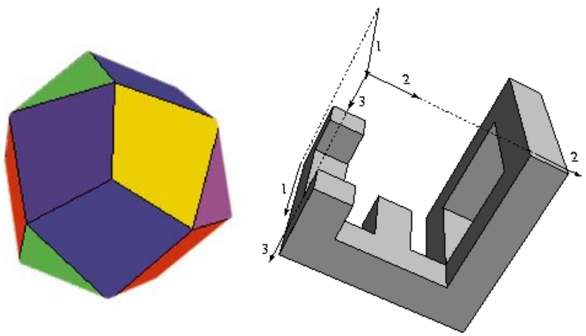
- A pinhole camera is a good approximation of the geometry of the imaging process
  - A true pinhole camera is a mathematical abstraction
- An **inverted** image is created on the image place in the "pinhole perspective projection model"
  - Every visible point in the scene can be traced via a straight line through the pinhole to the *image plane*



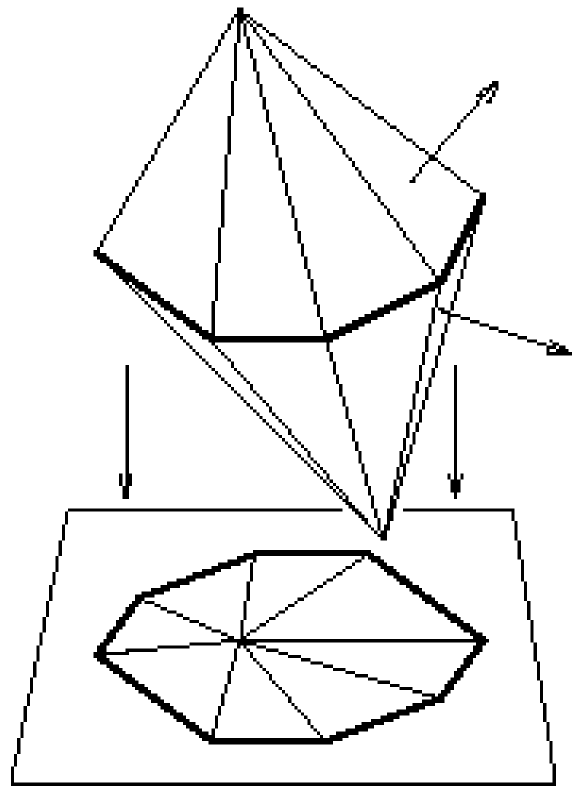
Perspective projection

- In such a **perspective projection**:
  - The image is inverted
    - Can instead consider an upright *virtual image*
  - Distant objects appear smaller in the image than near objects
  - Points project to points
  - Lines project to lines
  - What do circles project to?
  - What do parallel lines project to?
  - What do squares project to?
  - What do polyhedra project to?





Polyhedra



Polygons

Perspective and art

- Use of correct perspective projection indicated in 1st century B.C. frescoes
- Skill resurfaces in Renaissance: artists develop systematic methods to determine perspective projection (around 1480-1515)

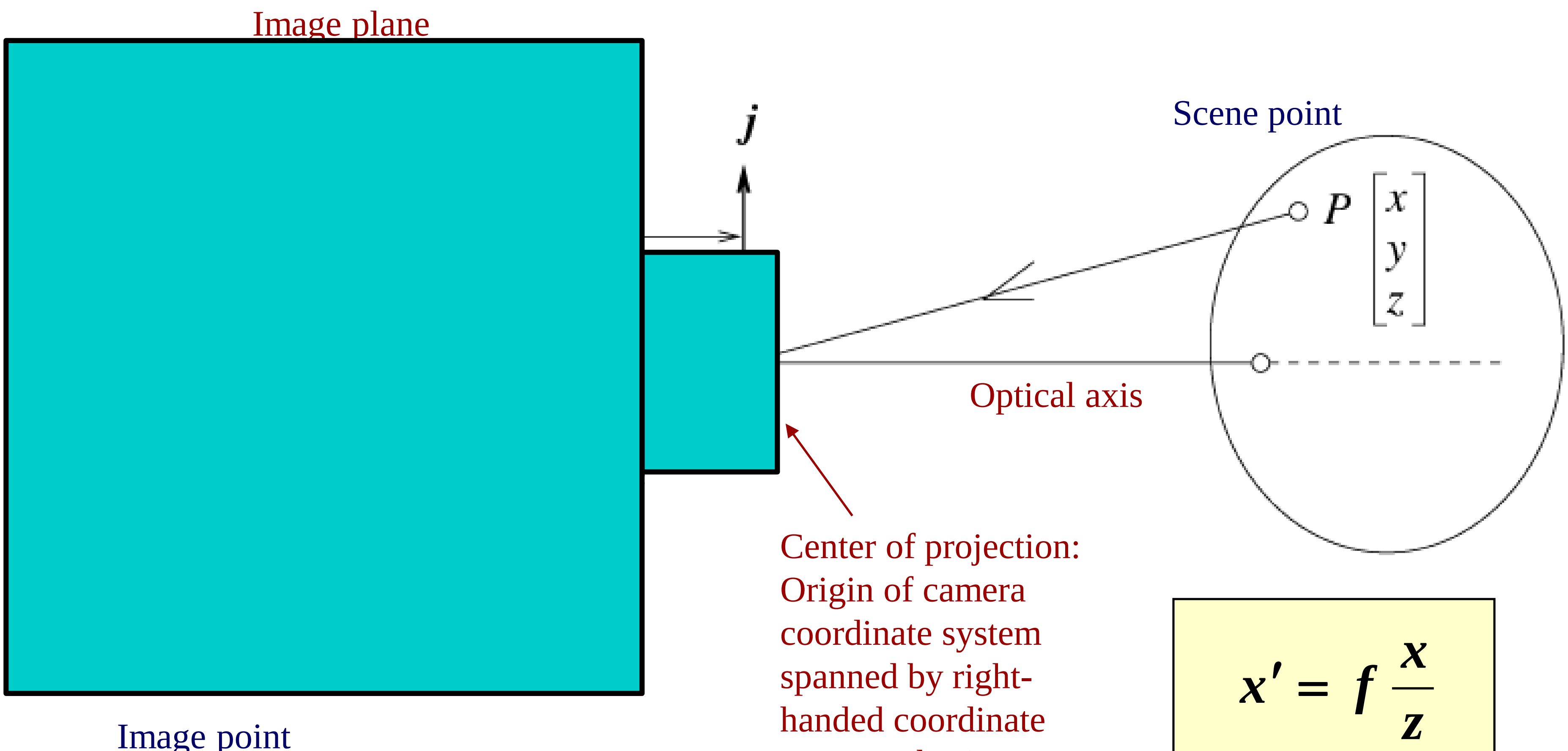
Raphael

Durer, 1525

Important parameters

- The origin of the camera coordinate system (*O*)
  - The center of projection
  - "Inside the lens"
- The optical axis
  - Perpendicular to the lens
  - Z axis
- The focal distance (*f*)
  - For now, this is the distance of the image plane Π' (along *z*)
- The image plane (Π')
  - Real or virtual
- The image center (*C'*)
  - Where the optical axis intersects the image plane

Pinhole perspective projection model



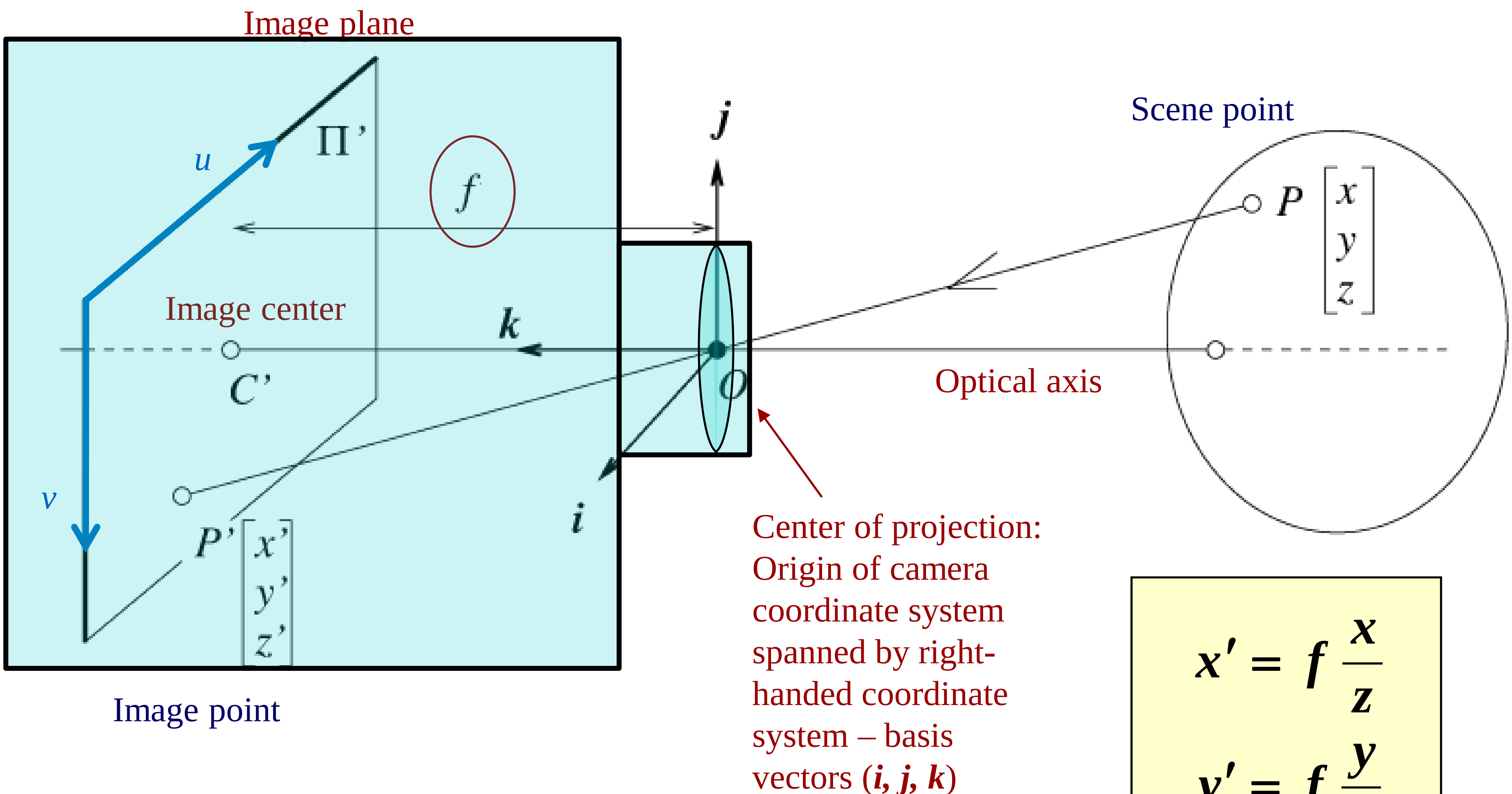
Center of projection:
Origin of camera coordinate system spanned by right-handed coordinate system – basis vectors (***i, j, k***)

$$x' = f\frac{x}{z}$$

$$y' = f\frac{y}{z}$$

$$z' = f$$

Pinhole perspective projection model



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

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

$$z' = f$$

Perspective Projection Matrix

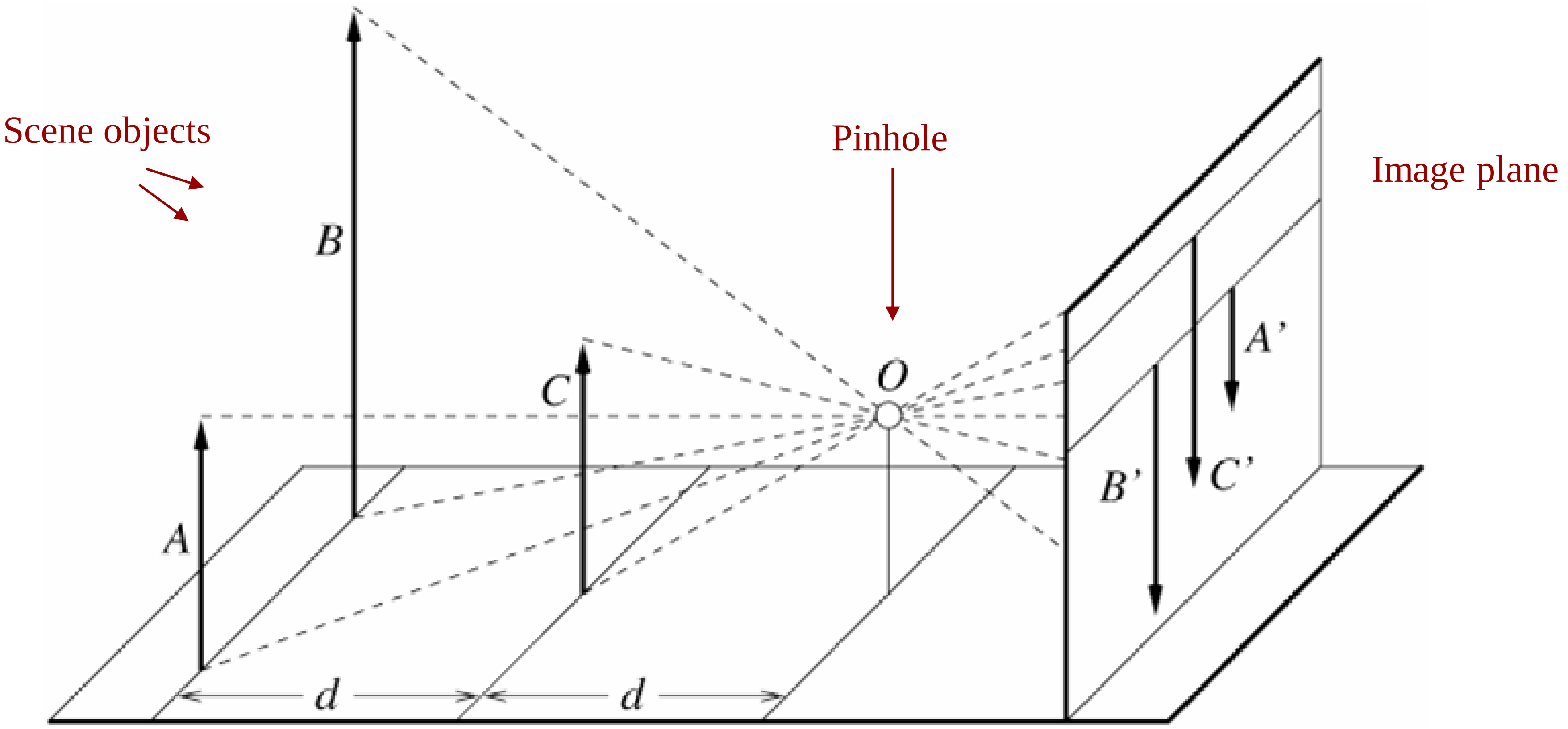
- Projection is a matrix multiplication using homogeneous coordinates:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f' & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z/f \end{bmatrix} \Rightarrow \left(f\frac{x}{z}, f\frac{y}{z}\right)$$

divide by the third coordinate to convert back to non-homogeneous coordinates

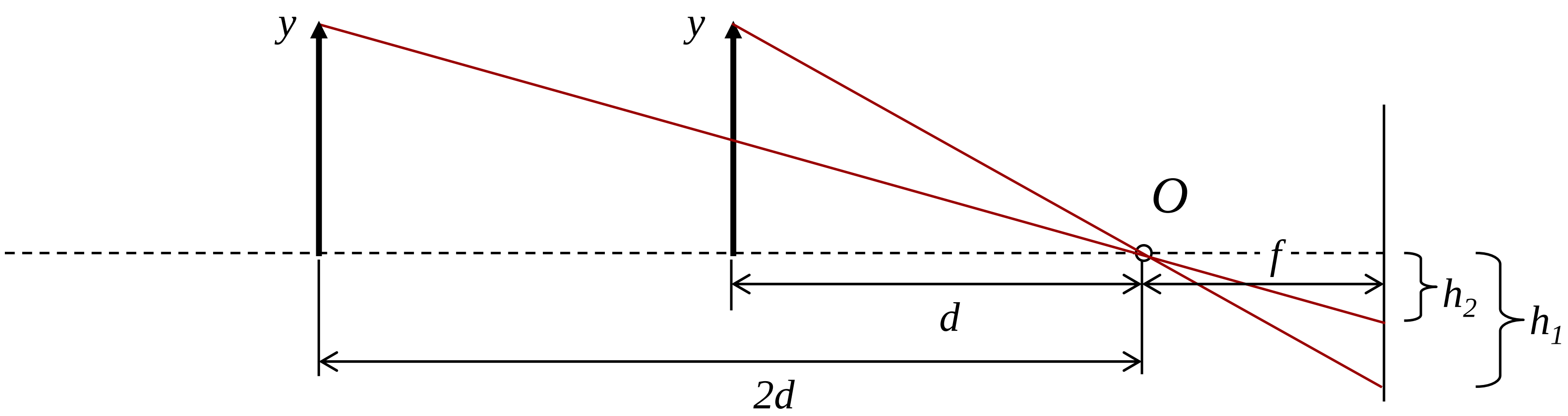
Complete mapping from world points to image pixel positions?

Projection geometry



Will the same object twice as far away look half the size?

Projection geometry

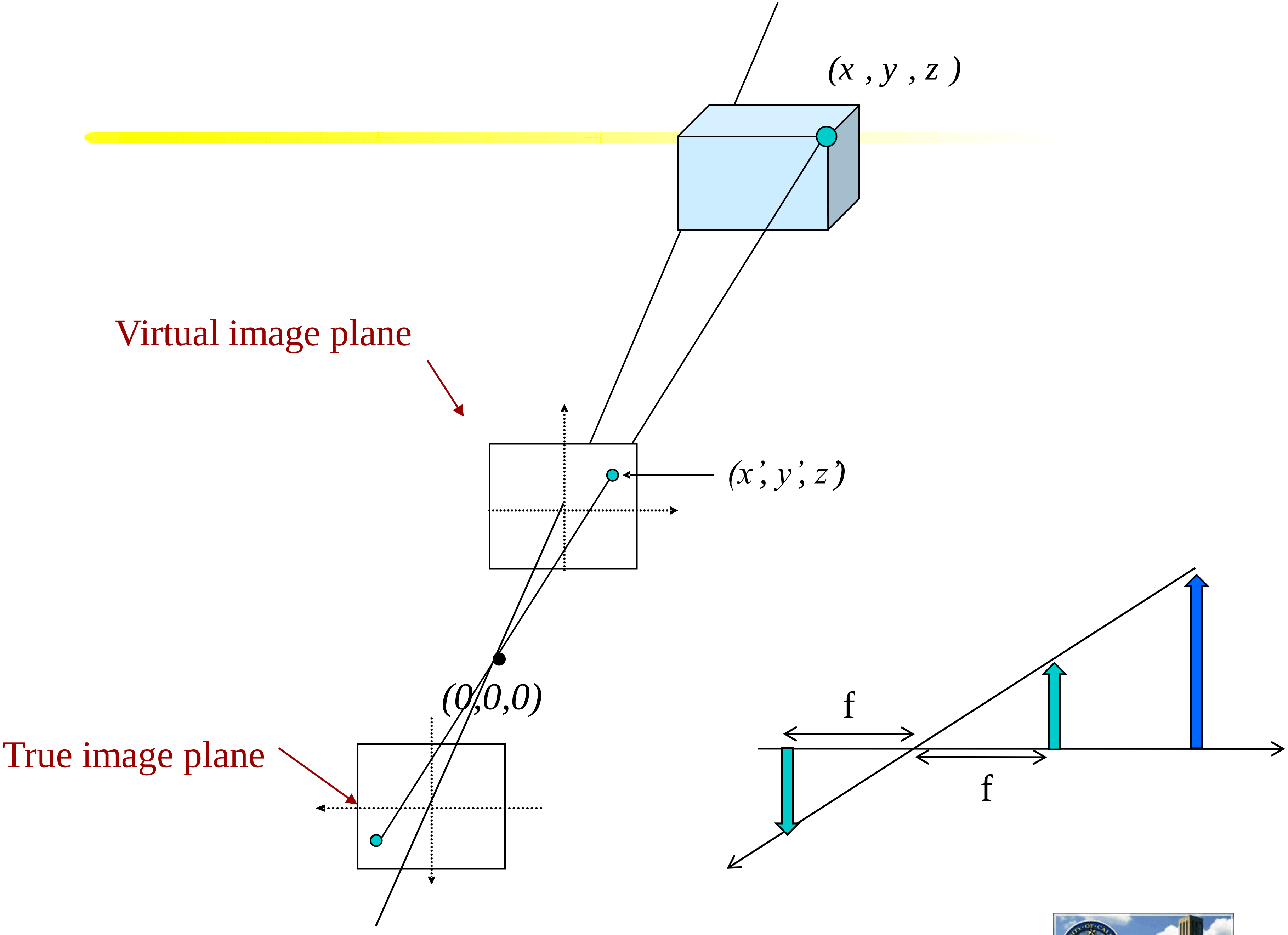


$$\frac{y}{d} = \frac{h_1}{f} \quad \text{and} \quad \frac{y}{2d} = \frac{h_2}{f}$$

So $h_1 = 2h_2$

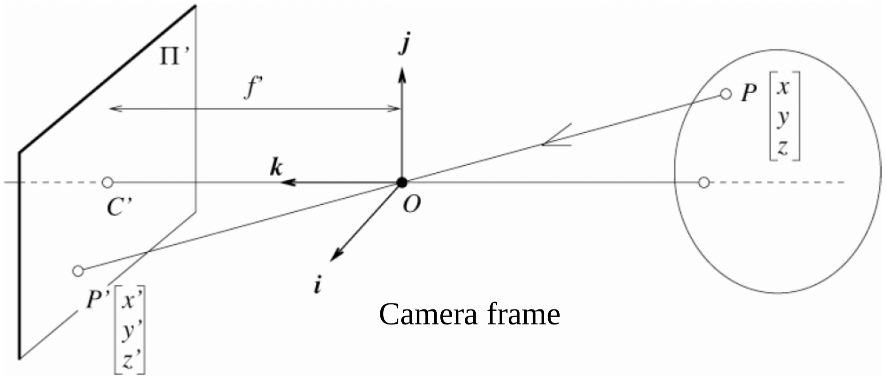
Virtual image plane

- Projection inverts the image, which can make the math inconvenient and the intuition difficult
  - It’s easier to think of the upright scene producing an upright image!
- So, we’ll often use the virtual image plane rather than the true image plane
  - The geometry is essentially the same, with sign reversals
- Caveat: When dealing with lens details and focusing, we’ll have to first deal with the true image plane

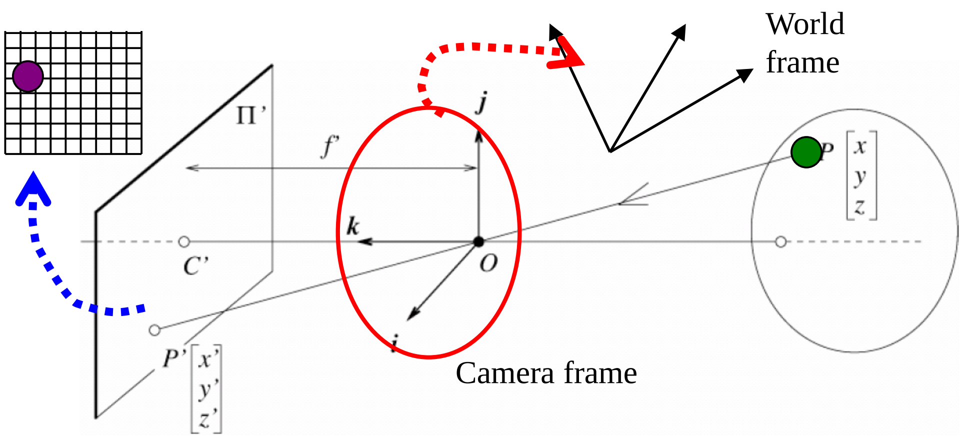

(x , y , z)
Virtual image plane
(x', y', z')
(0,0,0)
True image plane
f
f

Perspective projection & calibration

- Perspective equations so far in terms of *camera's* reference frame….
- Camera's *intrinsic* and *extrinsic* parameters needed to calibrate geometry.



Perspective projection & calibration



Extrinsic:

Camera frame ←→World frame

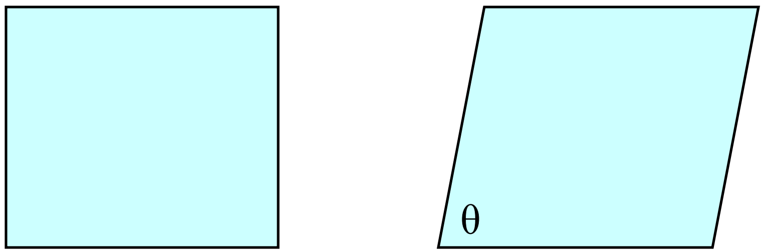
Intrinsic:

Image coordinates relative to camera ←→ Pixel coordinates

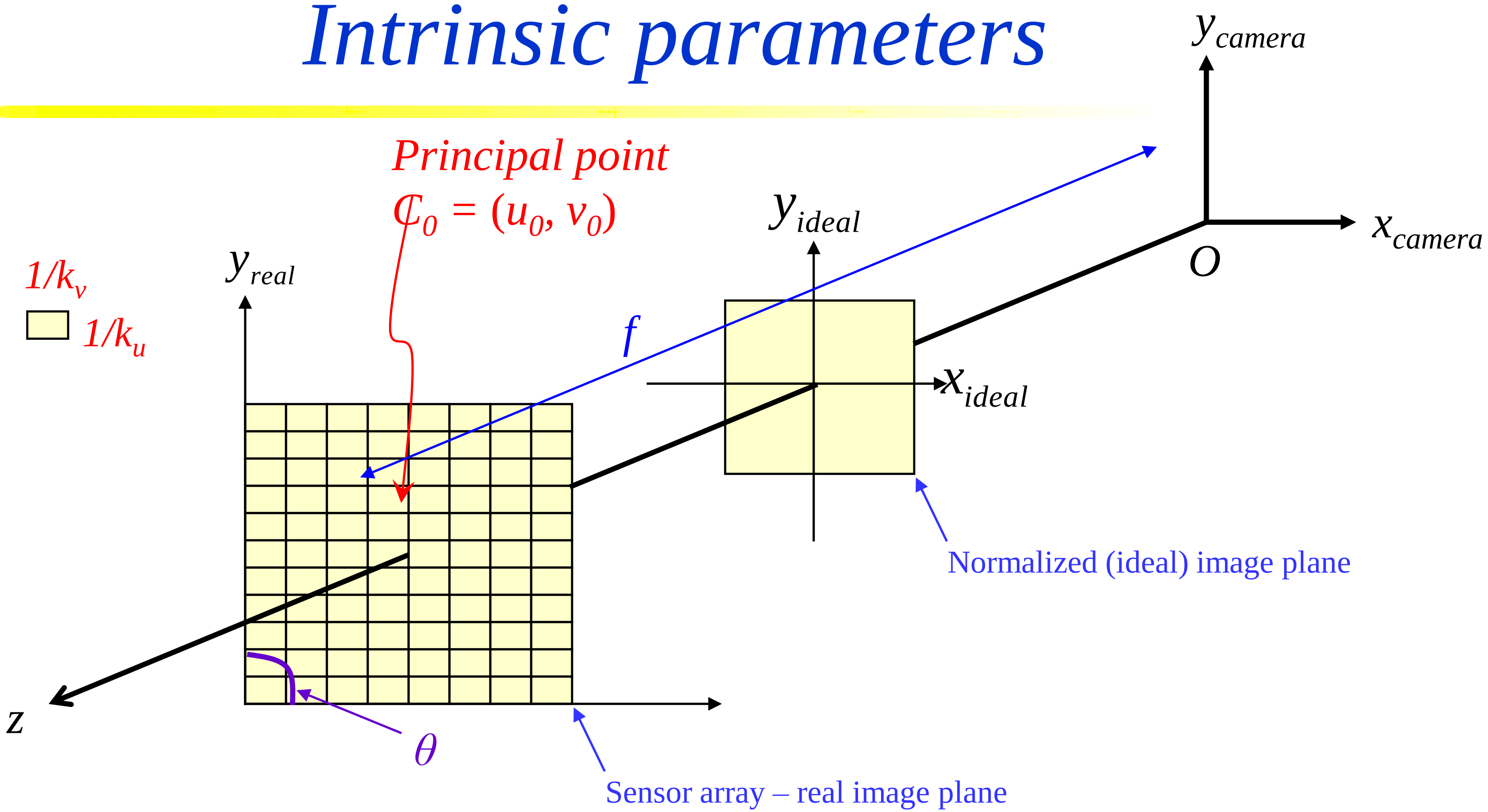
3D point (4x1)

Intrinsic parameters

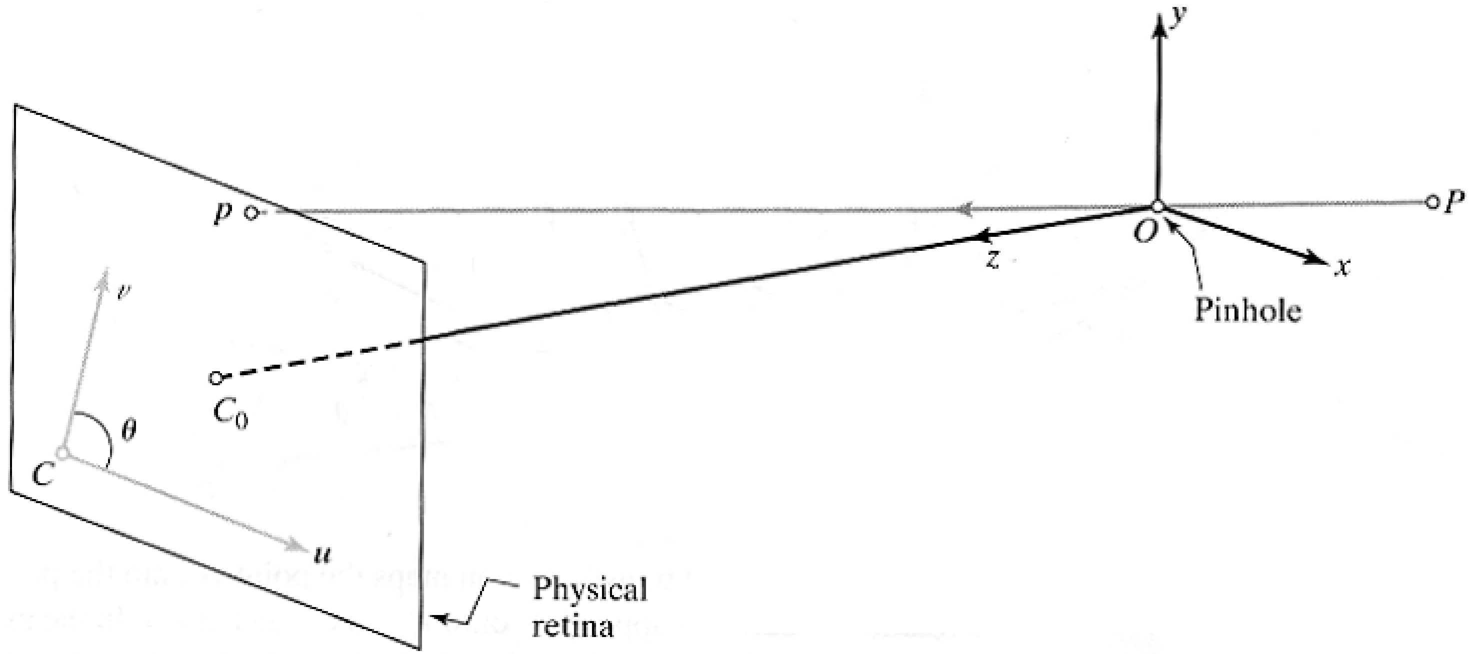
- 5 intrinsic parameters account for
  - The focal length (f)
  - The principal point $(C_0)=(u_0,v_0)$
    - Where the optical axis intersects the image plane
  - Pixel aspect ratio (k_u, k_v)
    - Pixels aren't necessarily square
  - Angle between the axes (θ)
    - Skewness in manufacturing



Intrinsic parameters



Intrinsic parameters: from idealized world coordinates to pixel values



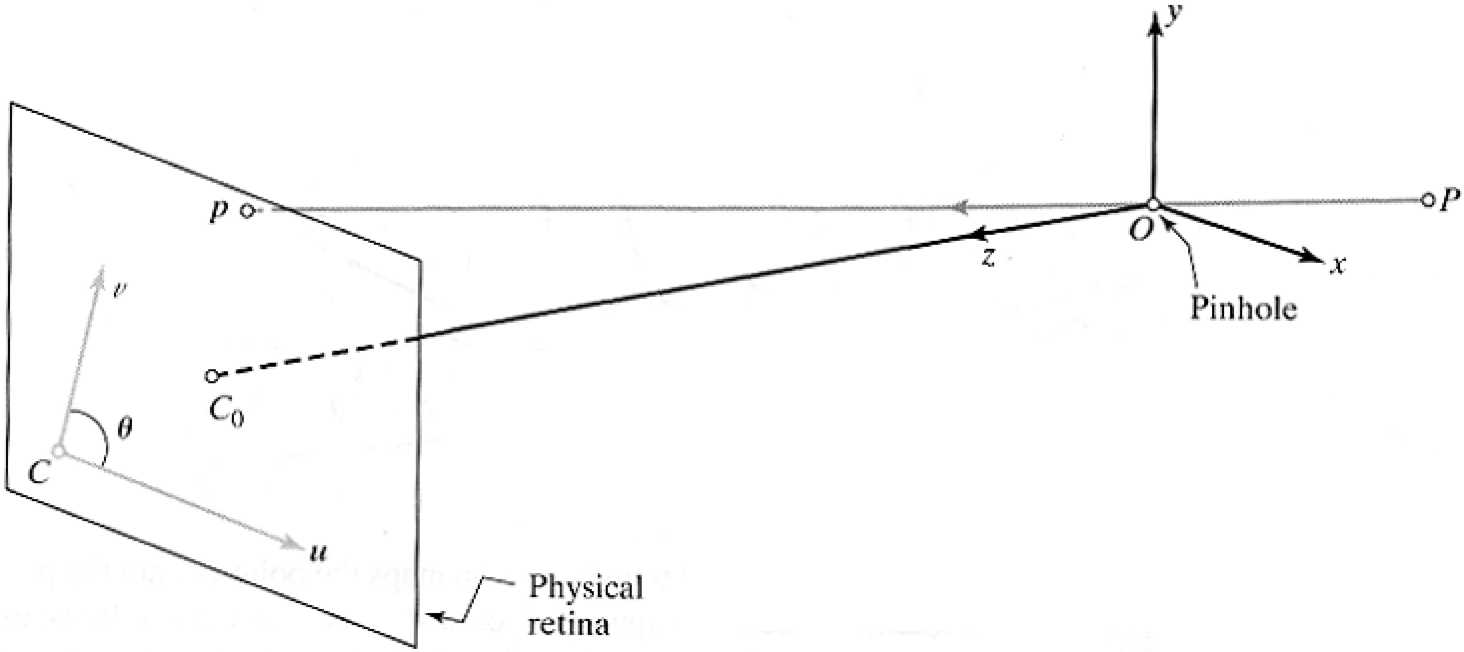
Perspective projection

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

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

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f' & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z/f \end{bmatrix}$$

Intrinsic parameters



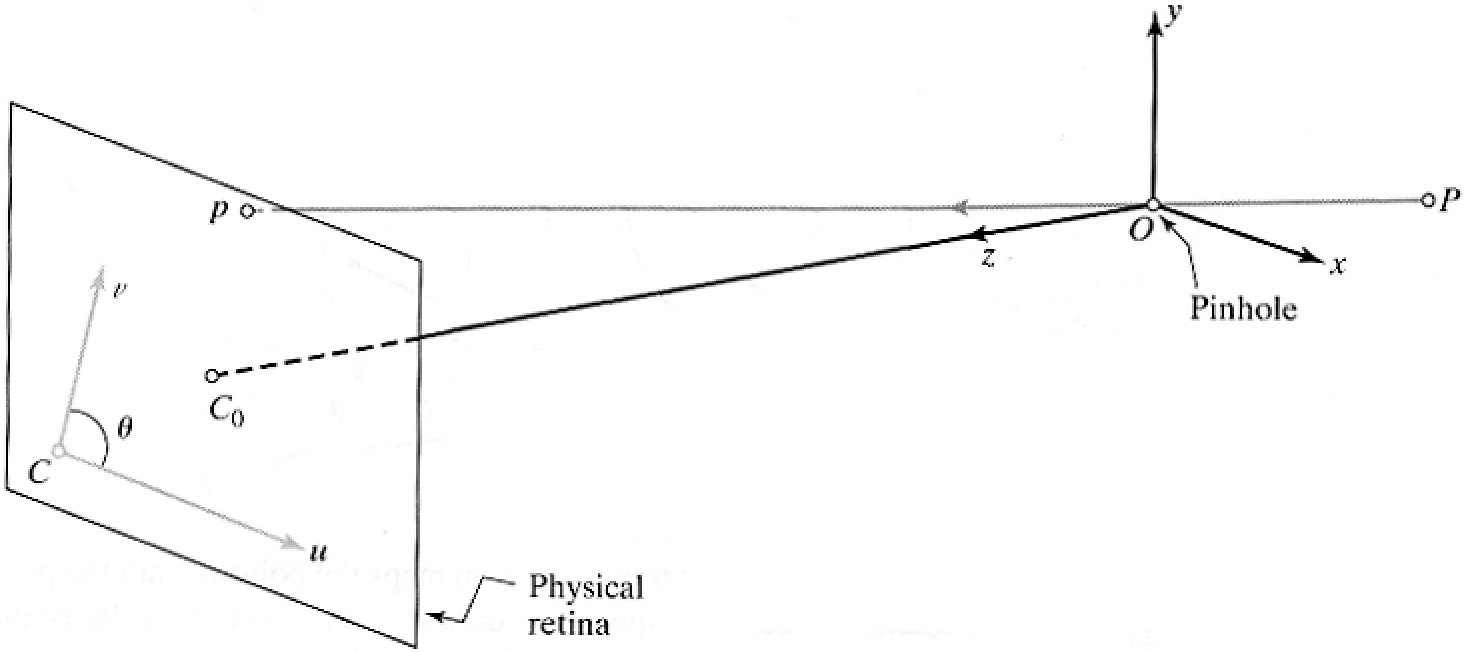
But “pixels” are in some arbitrary spatial units

$$u = \alpha \frac{x}{z}$$

$$v = \alpha \frac{y}{z}$$

W. Freeman

Intrinsic parameters



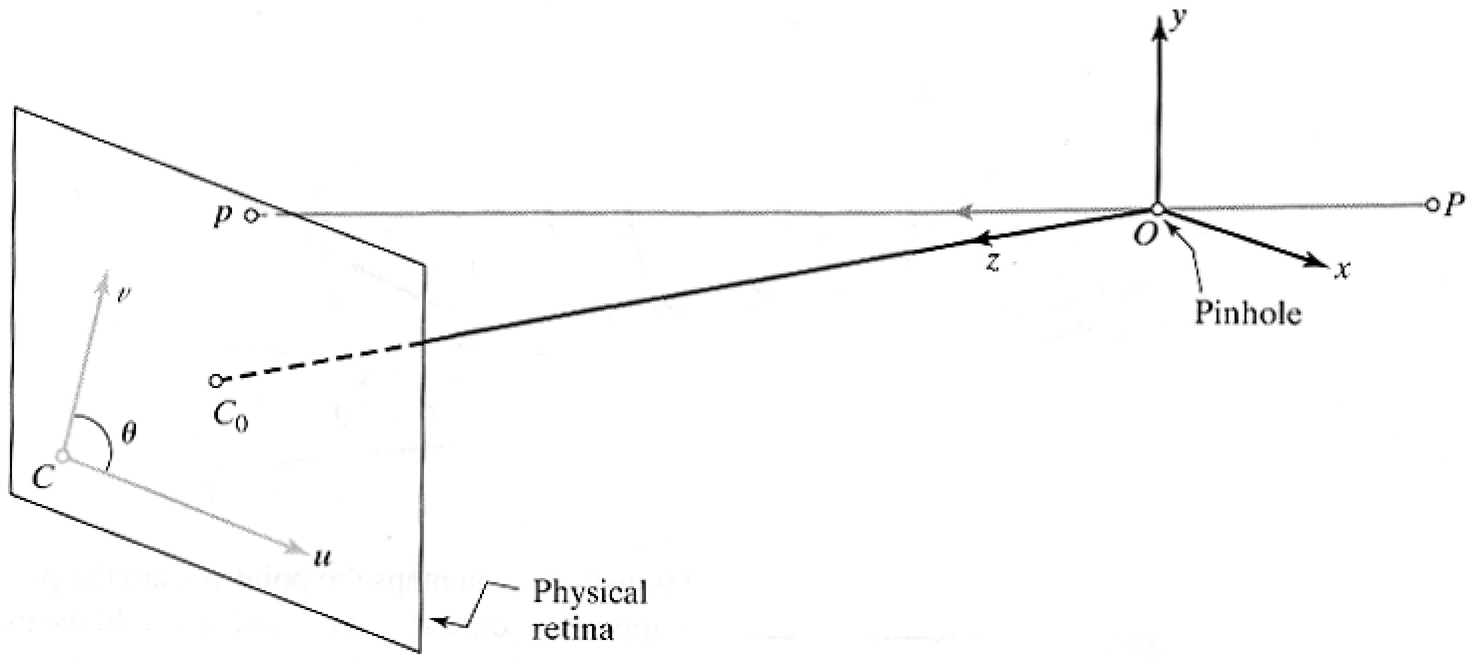
Maybe pixels are not square

$$u = \alpha \frac{x}{z}$$

$$v = \beta \frac{y}{z}$$

W. Freeman

Intrinsic parameters

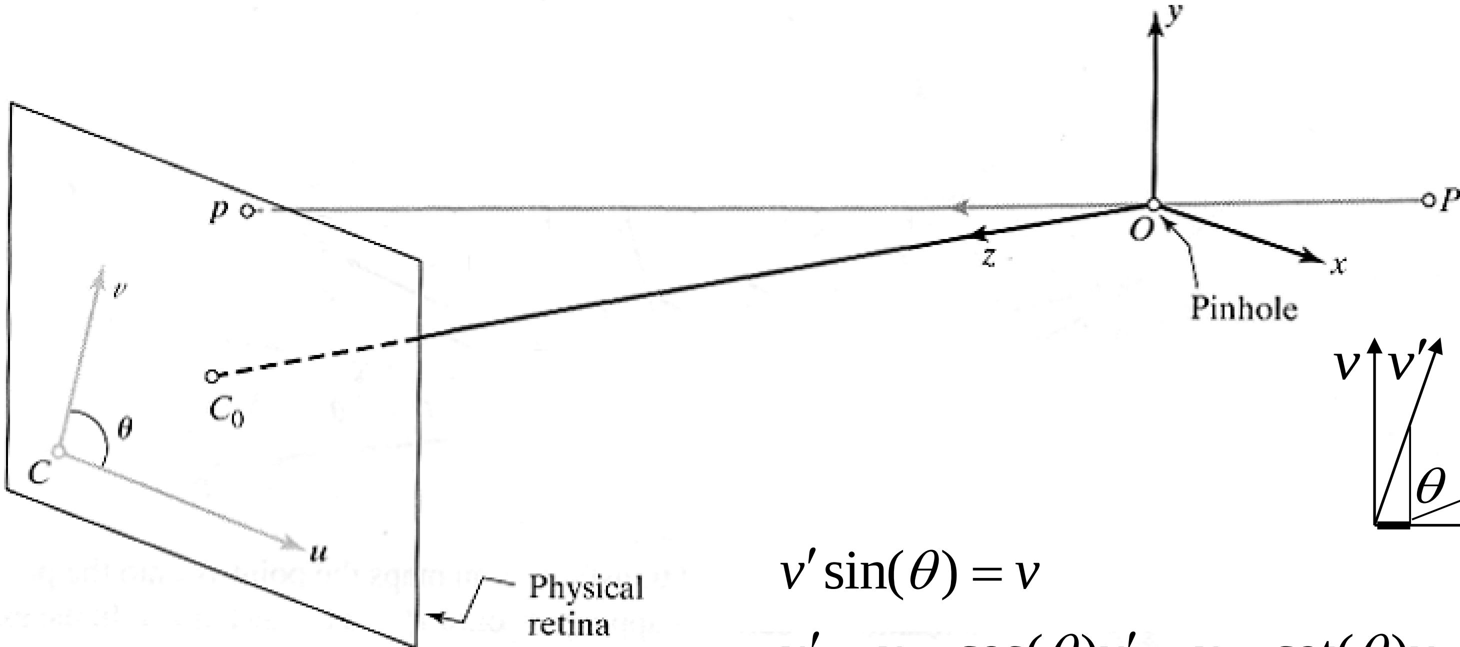


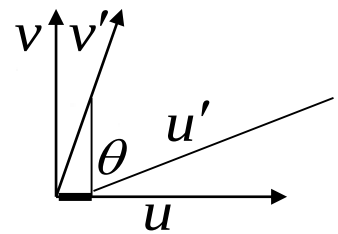
We don't know the origin of our camera pixel coordinates

$$u = \alpha \frac{x}{z} + u_0$$

$$v = \beta \frac{y}{z} + v_0$$

Intrinsic parameters





$$v' \sin(\theta) = v$$

$$u' = u - \cos(\theta) v' = u - \cot(\theta) v$$

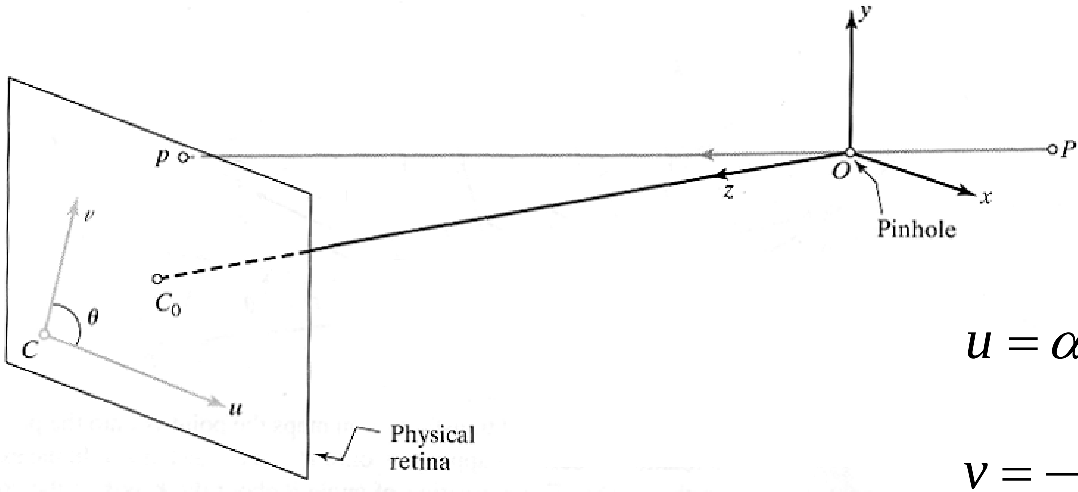
May be skew between camera pixel axes

$$u = \alpha \frac{x}{z} - \alpha \cot(\theta) \frac{y}{z} + u_0$$

$$v = \frac{\beta}{\sin(\theta)} \frac{y}{z} + v_0$$

W. Freeman

Intrinsic parameters, homogeneous coordinates



$$u = \alpha \frac{x}{z} - \alpha \cot(\theta) \frac{y}{z} + u_0$$

$$v = \frac{\beta}{\sin(\theta)} \frac{y}{z} + v_0$$

Using homogenous coordinates,

we can write this as:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha & -\alpha \cot(\theta) & u_0 & 0 \\ 0 & \dfrac{\beta}{\sin(\theta)} & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

or:

In pixels → $\vec{p} \quad = \quad \mathrm{K} \quad {}^{C}\vec{p}$ ← In camera-based coords

Extrinsic parameters: translation and rotation of camera frame

$$ {}^{C}\vec{p} = {}^{C}_{W}R \; {}^{W}\vec{p} + {}^{C}_{W}\vec{t} $$

Non-homogeneous coordinates

$$ \begin{pmatrix} {}^{C}\vec{p} \end{pmatrix} = \begin{pmatrix} - & - & - & | \\ - & {}^{C}_{W}R & - & {}^{C}_{W}\vec{t} \\ - & - & - & | \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} {}^{W}\vec{p} \end{pmatrix} $$

Homogeneous coordinates

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates

pixels → $\vec{p} = \mathrm{K}\ {}^{C}\vec{p}$ ← Camera coordinates

Intrinsic

World coordinates

$$
{}^{C}\vec{p} = \begin{pmatrix} {}^{C}_{W}R & {}^{C}_{W}\vec{t} \\ 0\ \ 0\ \ 0 & 1 \end{pmatrix} {}^{W}\vec{p}
$$

Extrinsic

$$
\vec{p} = K\begin{pmatrix} {}^{C}_{W}R & {}^{C}_{W}\vec{t} \end{pmatrix} {}^{W}\vec{p}
$$

$$
\vec{p} = M\ {}^{W}\vec{p}
$$

Homography

- Two special cases:
  - Object is a single plane
  - The camera execute a simple rotation
- Proof, based on projection equation, that image coordinates in multiple frames are related by homography

Other ways to write the same equation

pixel coordinates

world coordinates

$$\vec{p} = M \; {}^{W}\vec{p}$$

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} . & m_1^T & . & . \\ . & m_2^T & . & . \\ . & m_3^T & . & . \end{pmatrix} \begin{pmatrix} {}^{W}p_x \\ {}^{W}p_y \\ {}^{W}p_z \\ 1 \end{pmatrix}$$

Conversion back from homogeneous coordinates leads to:

$$\begin{cases} u = \dfrac{m_1 \cdot \vec{P}}{m_3 \cdot \vec{P}} \\ v = \dfrac{m_2 \cdot \vec{P}}{m_3 \cdot \vec{P}} \end{cases}$$

W. Freeman

Putting It All Together

$$\mathbf{p}_{real} = \mathbf{M}_{real \leftarrow ideal} \mathbf{M}_{ideal \leftarrow camera} \mathbf{M}_{camera \leftarrow world} \mathbf{P}_{world}$$

$$= \mathbf{M}_{real \leftarrow world} \mathbf{P}_{world} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \end{bmatrix} \mathbf{P}_{world}$$

$$x_{real} = \frac{q_{11} X_{world} + q_{12} Y_{world} + q_{13} Z_{world} + q_{14}}{q_{31} X_{world} + q_{32} Y_{world} + q_{33} Z_{world} + q_{34}}$$

$$y_{real} = \frac{q_{21} X_{world} + q_{22} Y_{world} + q_{23} Z_{world} + q_{24}}{q_{31} X_{world} + q_{32} Y_{world} + q_{33} Z_{world} + q_{34}}$$

Usage

- Governing equation

$$\mathbf{p}_{real} = \mathbf{M}_{real \leftarrow ideal}\mathbf{M}_{ideal \leftarrow camera}\mathbf{M}_{camera \leftarrow world}\mathbf{P}_{world} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \end{bmatrix} \mathbf{P}_{world}$$

- Off-line process
  - Given known 3D coordinates (landmarks) and their 2D projections, calculate q's
- On-line process
  - Given arbitrary 3D coordinates (first down at 30 yards) and q's, calculate 2D coordinates (where to draw the first-and-ten line)
  - Given arbitrary 2D coordinates (images of a vehicle) and q's, calculate 3D coordinates (where to aim the gun to fire)

Camera Calibration and Registration

- First step
  - Estimate the combined transformation matrix $\mathbf{M}_{real \leftarrow world}$
- Second step
  - Estimate intrinsic camera parameters
  - Estimate extrinsic camera parameters
- Solution
  - Using objects of known sizes and shapes (6 points at least)
  - Each point provides two constraints (x,y)
  - A checked board pattern placed at different depths

Putting It All Together

$$\mathbf{p}_{real} = \begin{bmatrix} k_u & 0 & u_o \\ 0 & k_v & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ 0 & 1 \end{bmatrix} \mathbf{P}_{world}$$

$$\begin{bmatrix} k_u & 0 & u_o \\ 0 & k_v & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{r}_1 & t_x \\ \mathbf{r}_2 & t_y \\ \mathbf{r}_3 & t_z \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \alpha_u & 0 & u_o & 0 \\ 0 & \alpha_v & v_o & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{r}_1 & t_x \\ \mathbf{r}_2 & t_y \\ \mathbf{r}_3 & t_z \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha_u \mathbf{r}_1 + u_o \mathbf{r}_3 & \alpha_u t_x + u_o t_z \\ \alpha_v \mathbf{r}_2 + v_o \mathbf{r}_3 & \alpha_v t_y + v_o t_z \\ \mathbf{r}_3 & t_z \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{q}_1{}^T & q_{14} \\ \mathbf{q}_2{}^T & q_{24} \\ \mathbf{q}_3{}^T & q_{34} \end{bmatrix}$$

$$|\mathbf{q}_3| = 1, (\mathbf{q}_1 \times \mathbf{q}_3) \cdot (\mathbf{q}_2 \times \mathbf{q}_3) = 0$$

Camera Calibration

- Certainly, not all 3x4 matrices are like above
  - 3x4 matrices have 11 free parameters (with a scale factor that cannot be decided uniquely)
  - matrix in the previous slide has 10 parameters (2 scale, 2 camera center, 3 translation, 3 rotation)
  - additional constraints can be very useful
    - to solve for the matrix, and
    - to compute the parameters
  - Theorem: 3x4 matrices can be put in the form of the previous slide if and only if the following two constraints are satisfied

$$|\mathbf{q}_3| = 1, (\mathbf{q}_1 \times \mathbf{q}_3) \cdot (\mathbf{q}_2 \times \mathbf{q}_3) = 0$$

Finding the transform matrix

$$\mathbf{p}_{real} = \begin{bmatrix} \mathbf{q_1}^T & q_{14} \\ \mathbf{q_2}^T & q_{24} \\ \mathbf{q_3}^T & q_{34} \end{bmatrix} \mathbf{P}_{world}$$

$$\begin{bmatrix} wx_{real} \\ wy_{real} \\ w \end{bmatrix} = \begin{bmatrix} \mathbf{q_1}^T & q_{14} \\ \mathbf{q_2}^T & q_{24} \\ \mathbf{q_3}^T & q_{34} \end{bmatrix} \begin{bmatrix} X_{world} \\ Y_{world} \\ Z_{world} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{q_1}^T & q_{14} \\ \mathbf{q_2}^T & q_{24} \\ \mathbf{q_3}^T & q_{34} \end{bmatrix} \begin{bmatrix} \mathbf{P}^3_{world} \\ 1 \end{bmatrix}$$

$$\frac{\mathbf{q_1}^T \mathbf{P}^3_{world} + q_{14}}{\mathbf{q_3}^T \mathbf{P}^3_{world} + q_{34}} = x_{real} \Rightarrow \mathbf{q_1}^T \mathbf{P}^3_{world} - u\mathbf{q_3}^T \mathbf{P}^3_{world} + q_{14} - uq_{34} = 0$$

$$\frac{\mathbf{q_2}^T \mathbf{P}^3_{world} + q_{24}}{\mathbf{q_3}^T \mathbf{P}^3_{world} + q_{34}} = y_{real} \Rightarrow \mathbf{q_2}^T \mathbf{P}^3_{world} - v\mathbf{q_3}^T \mathbf{P}^3_{world} + q_{24} - vq_{34} = 0$$

$$\Rightarrow \mathbf{AQ} = \mathbf{0}$$

Finding the transform matrix (cont.)

- Each data point provide two equations, with at least 6 points we will have 12 equations for solving 11 numbers up to a scale factor
- Lagrange multipliers can be used to incorporate other constraints
  - The usual constraint is $\mathbf{q}_3^2=1$
- Afterward, both intrinsic and extrinsic parameters can be recovered

Details

$$\mathbf{AQ} = \mathbf{0}$$

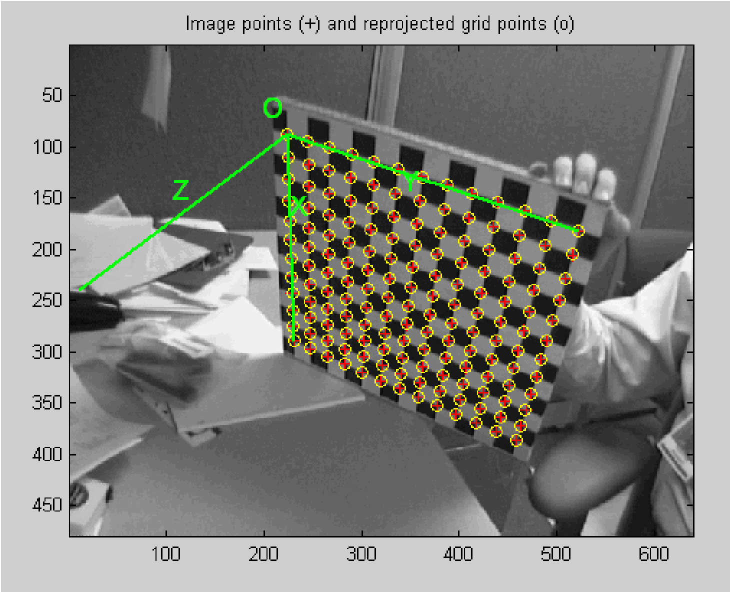
$$\min \|\mathbf{AQ}\|^2 \text{ subject to } |\mathbf{q}_3| = 1$$

$$\min \|\mathbf{AQ}\|^2 + \lambda(1 - |\mathbf{q}_3|^2)$$

- Solved by Langrage multiplier

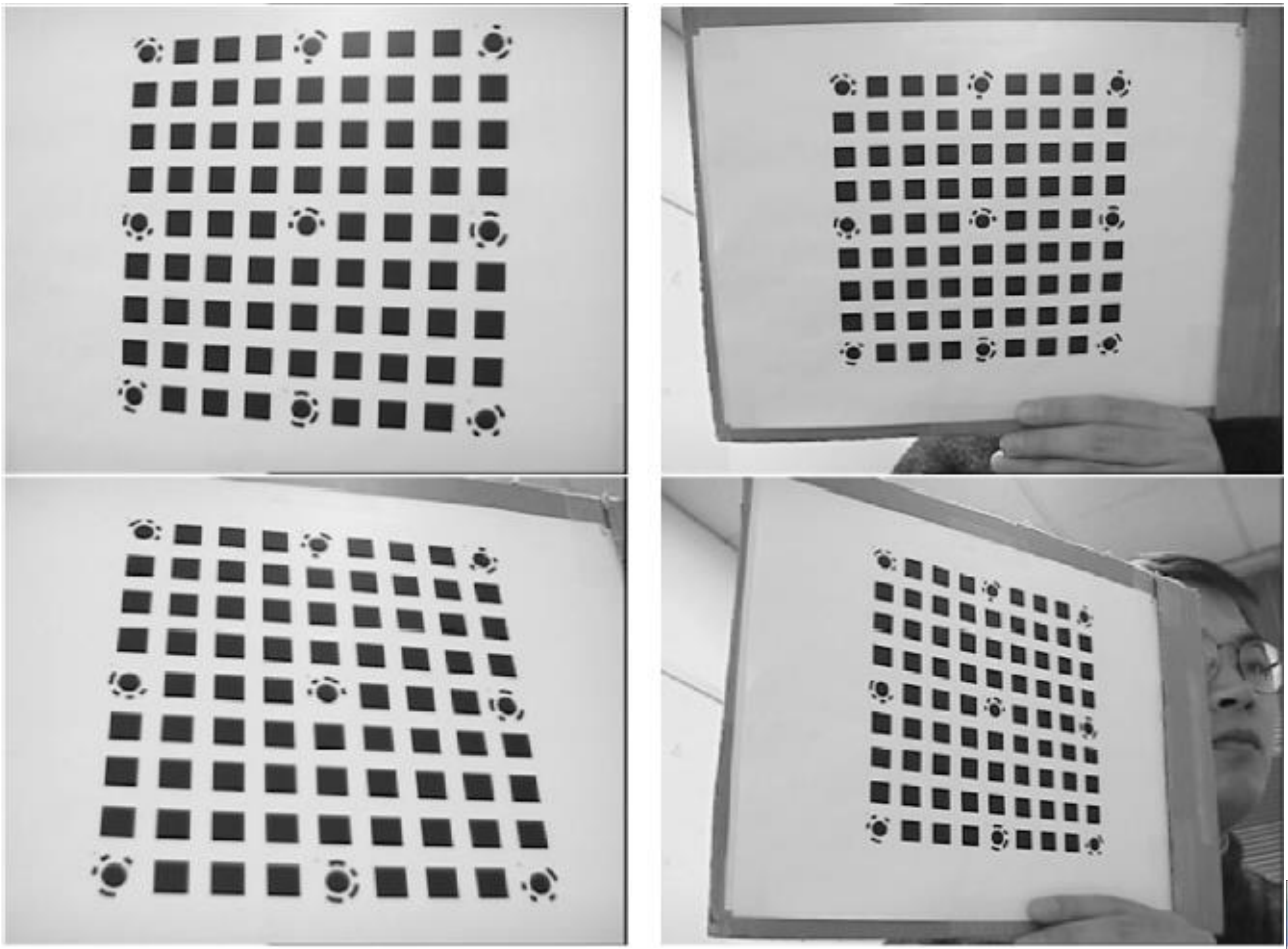
Calibration software available

- Lots of it these days!
- E.g., Camera Calibration Toolkit for Matlab
  - From the Computational Vision Group at Caltech (Bouguet)
    - http://www.vision.caltech.edu/bouguetj/calib_doc/
    - Includes lots of links to calibration tools and research



Flexible Pattern Placement

- http://research.microsoft.com/~zhang/calib/



Step 1: Intrinsic Parameters

- $\mathbf{x}_{image}=\mathbf{H}\mathbf{x}_{plane}=\mathbf{K}[\mathbf{R}_1,\mathbf{R}_2,\mathbf{T}]\mathbf{x}_{plane}$

$$\mathbf{H}=\mathbf{K}[\mathbf{R}_1\mathbf{R}_2\mathbf{T}]$$
$$\Rightarrow [\mathbf{h}_1 \quad \mathbf{h}_2 \quad \mathbf{h}_3]=\mathbf{K}[\mathbf{R}_1\mathbf{R}_2\mathbf{T}]$$
$$\Rightarrow [\mathbf{K}^{-1}\mathbf{h}_1 \quad \mathbf{K}^{-1}\mathbf{h}_2 \quad \mathbf{K}^{-1}\mathbf{h}_3]=[\mathbf{R}_1\mathbf{R}_2\mathbf{T}]$$
$$\Rightarrow \begin{matrix} (\mathbf{K}^{-1}\mathbf{h}_1)^T\mathbf{K}^{-1}\mathbf{h}_2=0 & (\mathbf{K}^{-1}\mathbf{h}_1)^T\mathbf{K}^{-1}\mathbf{h}_1=(\mathbf{K}^{-1}\mathbf{h}_1)^T\mathbf{K}^{-1}\mathbf{h}_1 \\ \mathbf{h}_1{}^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{h}_2=0 & \mathbf{h}_1{}^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{h}_1=\mathbf{h}_2{}^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{h}_2 \end{matrix}$$

Image of absolute conic

- One homography
  - 8 DOFs
  - 6 extrinsic parameters (3 rotation + 3 translation)
  - 2 constraints on intrinsic parameters
  - 3 planes in general configuration

Step 2: Extrinsic Parameters

$$\mathbf{H} = \mathbf{K}[\mathbf{R}_1 \mathbf{R}_2 \mathbf{T}]$$

$$\Rightarrow \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \mathbf{h}_3 \end{bmatrix} = \mathbf{K}[\mathbf{R}_1 \ \mathbf{R}_2 \ \mathbf{T}]$$

$$\Rightarrow \mathbf{R}_1 = \lambda \mathbf{K}^{-1}\mathbf{h}_1$$

$$\mathbf{R}_2 = \lambda \mathbf{K}^{-1}\mathbf{h}_2$$

$$\mathbf{T} = \lambda \mathbf{K}^{-1}\mathbf{h}_3$$

$$\mathbf{R}_3 = \mathbf{R}_1 \times \mathbf{R}_2$$

$$\lambda = \frac{1}{\left\|\mathbf{K}^{-1}\mathbf{h}_1\right\|} = \frac{1}{\left\|\mathbf{K}^{-1}\mathbf{h}_2\right\|}$$

Step 3: Intrinsic + Extrinsic Parameters

- $\mathbf{x}_{\text{image}} = \mathbf{H}\mathbf{x}_{\text{plane}} = \mathbf{K}[\mathbf{R}_1, \mathbf{R}_2, \mathbf{T}]\mathbf{x}_{\text{plane}}$

$$\sum_{i=1}^{n} \sum_{j=1}^{m} (\mathbf{x}_{ij}^{\mathbf{image}} - \mathbf{K}[\mathbf{R}_1 \mathbf{R}_2 \mathbf{T}]\mathbf{x}_{ij}^{\text{model}})^2$$

# of images   # of points/images

- Nonlinear optimization
- Using Lenvenberg-Marquardt in Minpack
- **K** from the previous step as initial guess

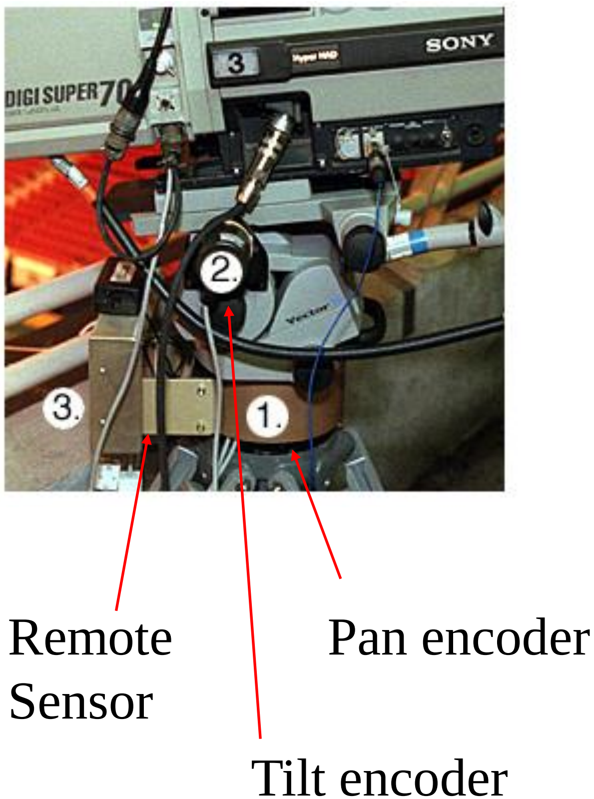
Usage

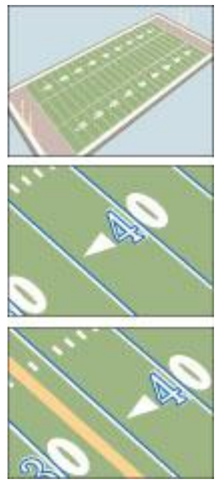
- Governing equation

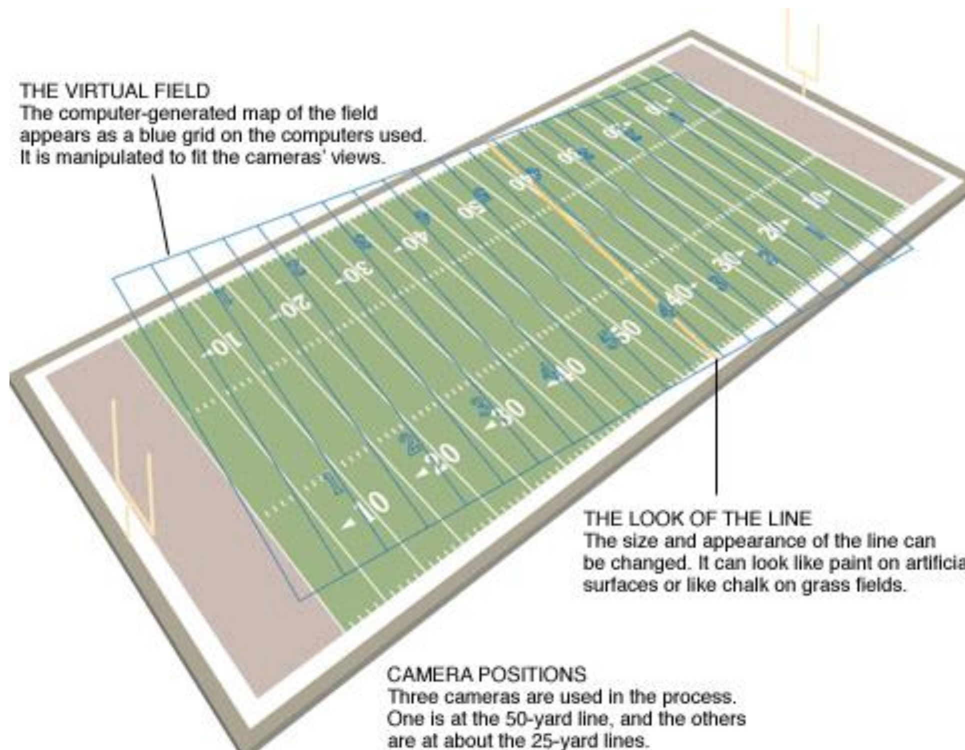
$$\mathbf{p}_{real} = \mathbf{M}_{real \leftarrow ideal} \mathbf{M}_{ideal \leftarrow camera} \mathbf{M}_{camera \leftarrow world} \mathbf{P}_{world} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \end{bmatrix} \mathbf{P}_{world}$$

- Off-line process
  - Given known 3D coordinates (landmarks) and their 2D projections, calculate q's
- On-line process
  - Given arbitrary 3D coordinates (first down at 30 yards) and q's, calculate 2D coordinates (where to draw the first-and-ten line)
  - Given arbitrary 2D coordinates (images of a vehicle) and q's, calculate 3D coordinates (where to aim the gun to fire)

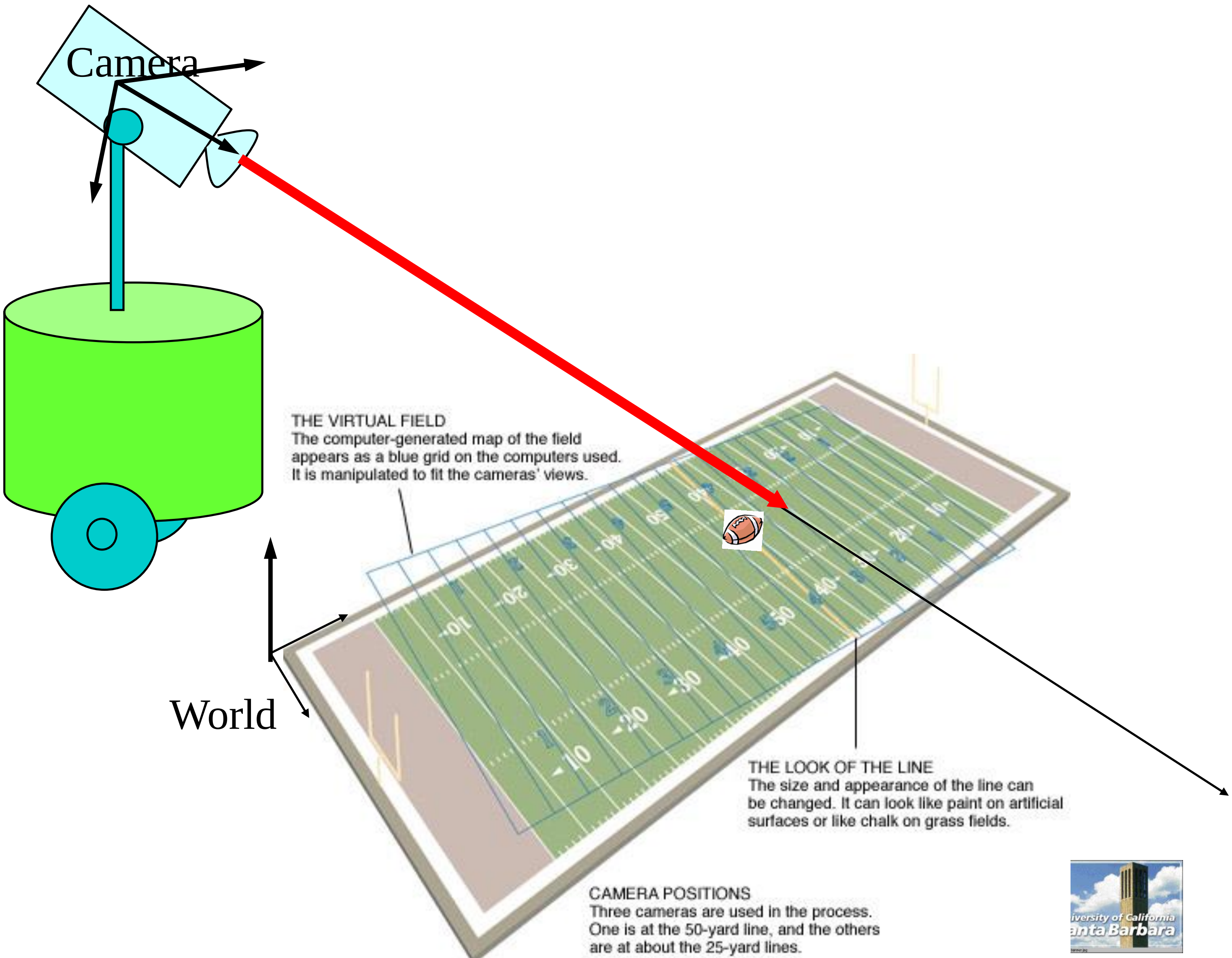
Mapping Contours



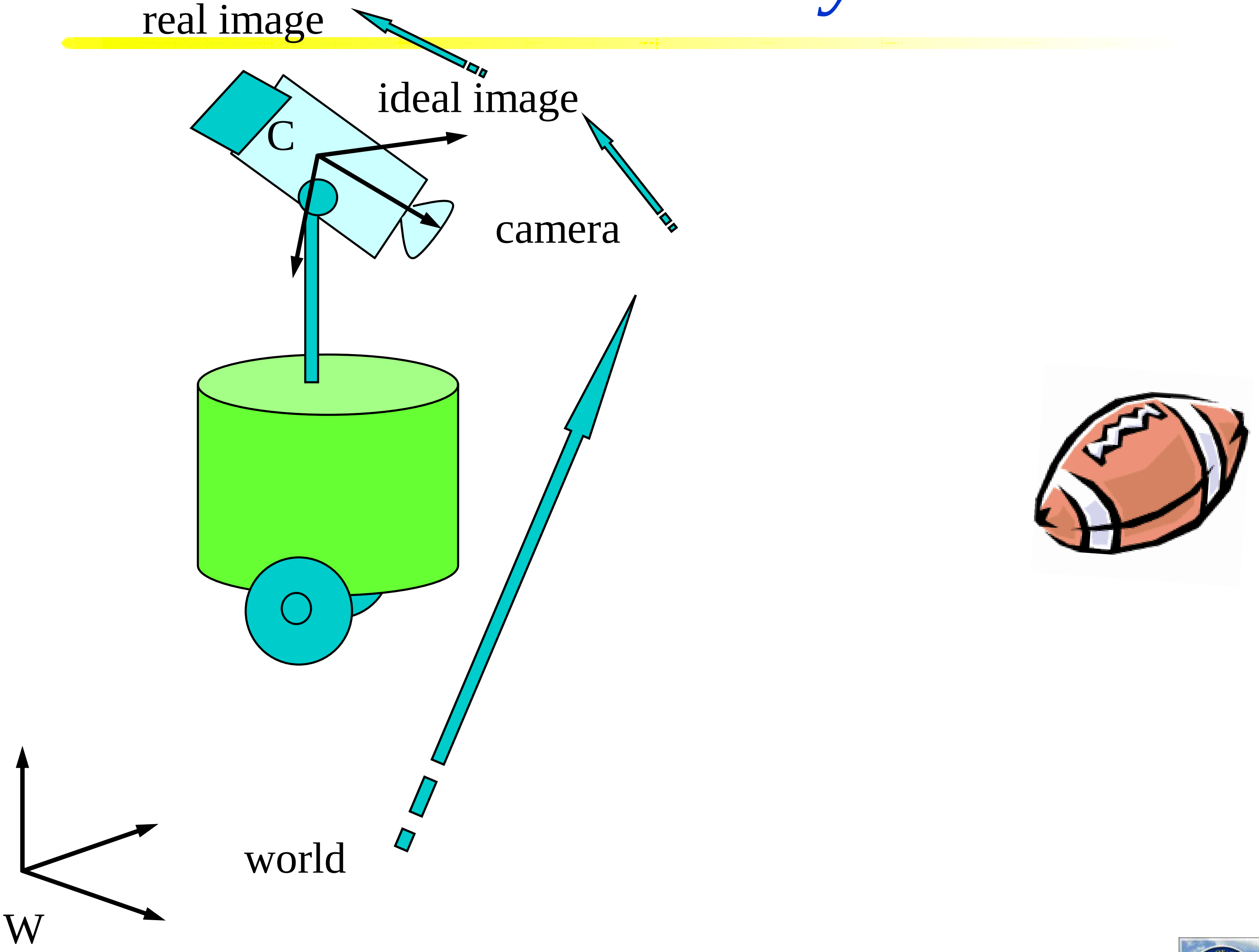




Coordinate systems



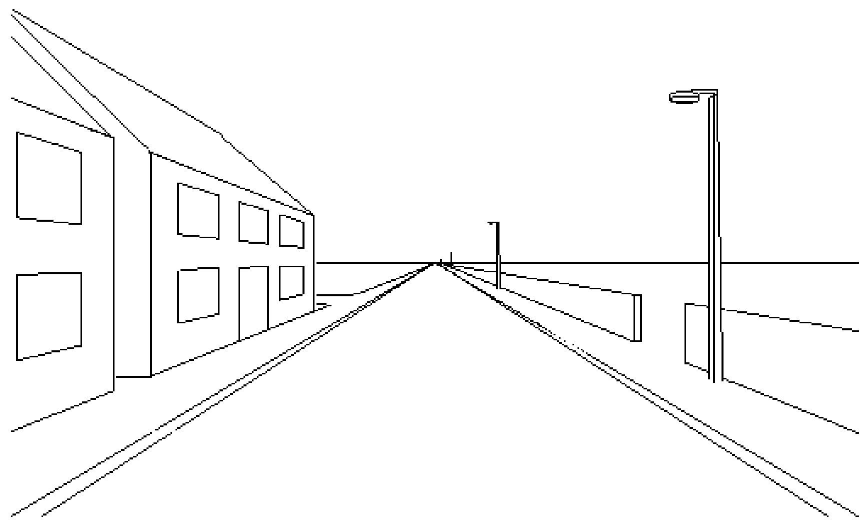
Coordinate systems



Example

Vanishing points, horizon lines

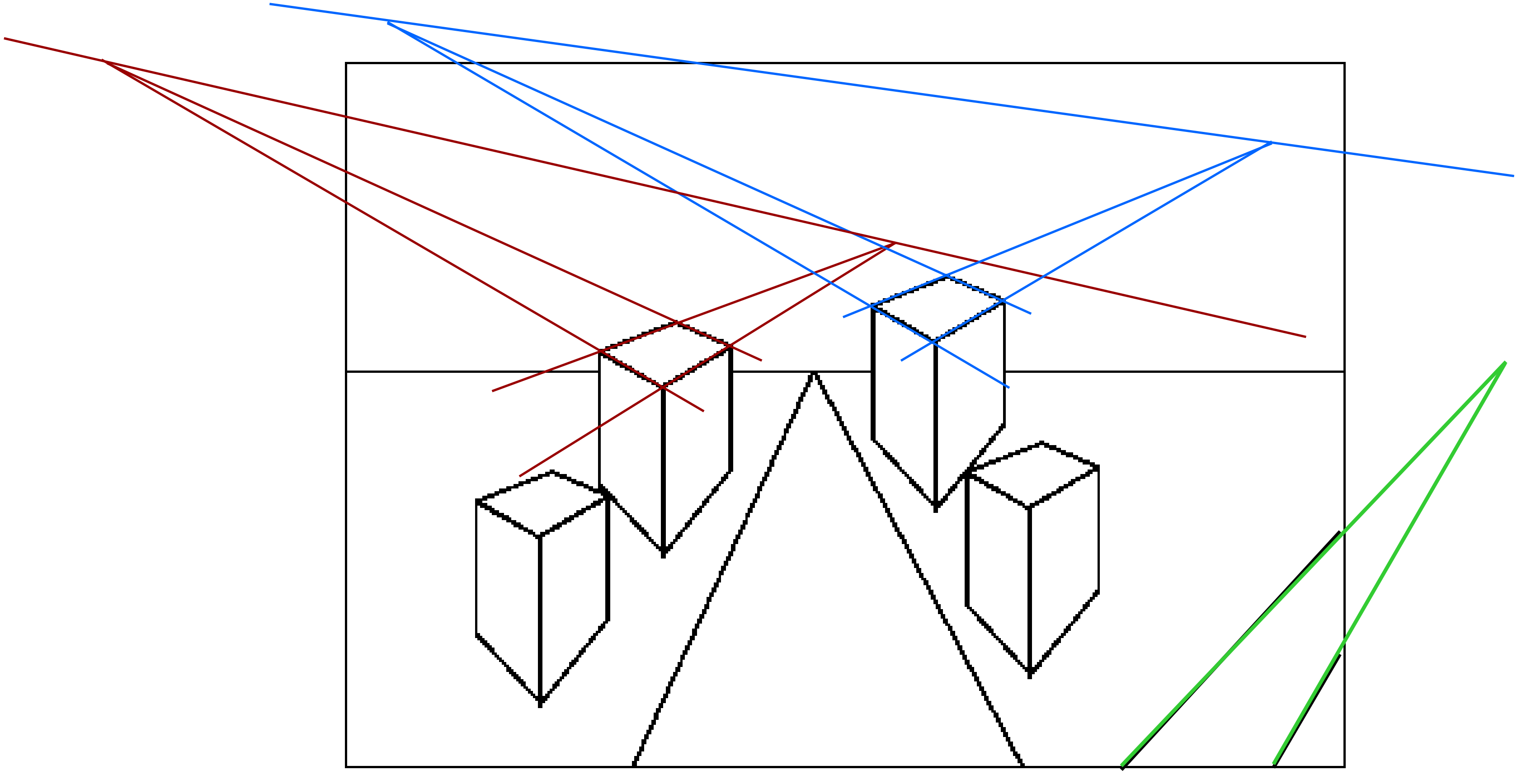
- Parallel lines in the scene intersect at the *horizon line*
  - Each pair of parallel lines meet at a *vanishing point*
  - The collection of vanishing points for all sets of parallel lines *in a given plane* is collinear, called the *horizon line* for that plane



Perspective effects

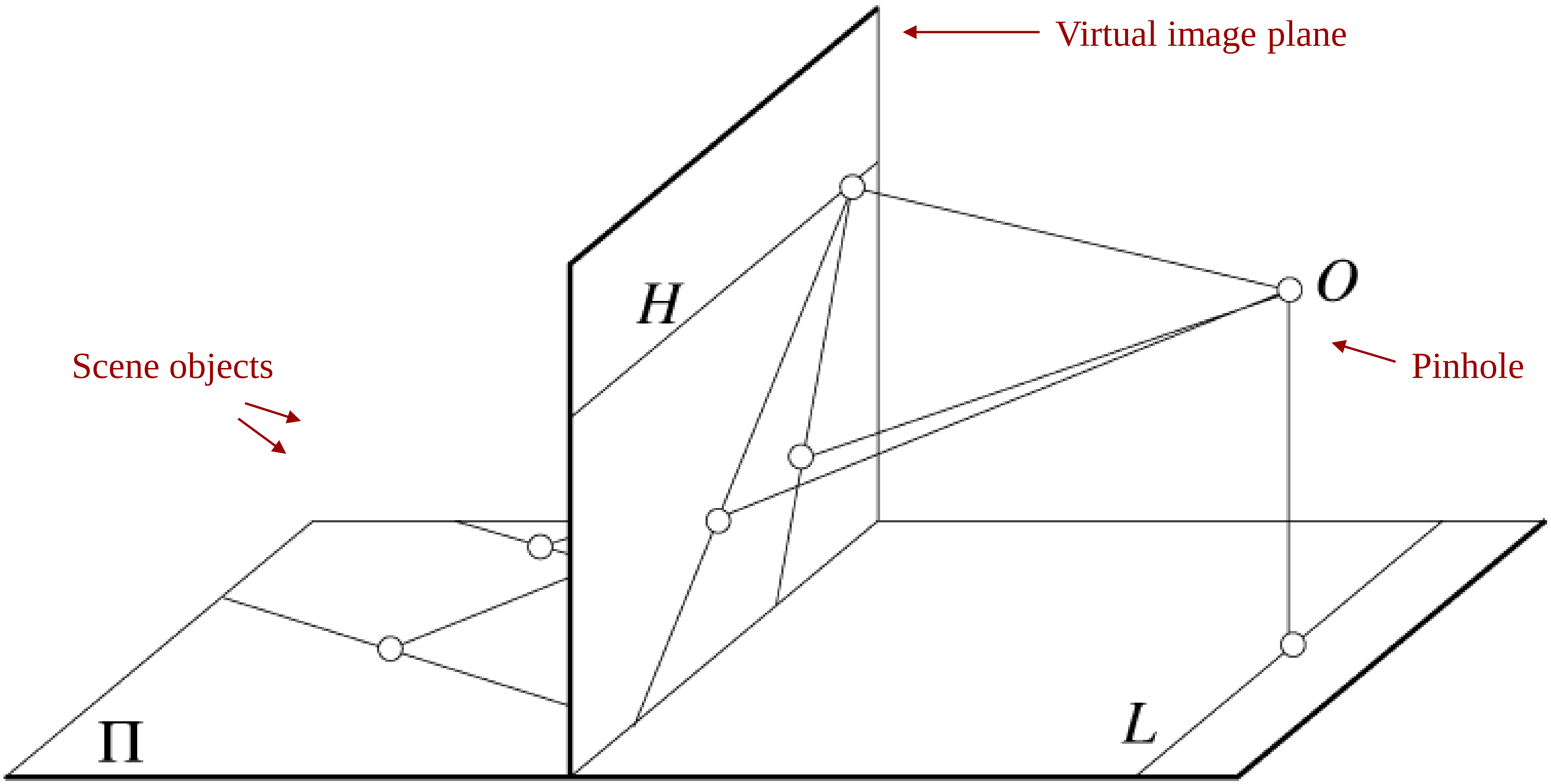
Example

Vanishing points, horizon lines



University of California Santa Barbara

Parallel lines



Where does the horizon line appear in the image?

Affine projection models

- Perspective projection is an ideal abstraction, an approximation of the true imaging geometry
- There are other projection models too!
- **Affine projection models** are simpler, though less accurate:
  - Orthographic projection
  - Parallel projection
  - Weak-perspective projection
  - Paraperspective projection
  - No matter which model you use, the equations are **linear**

Affine Camera

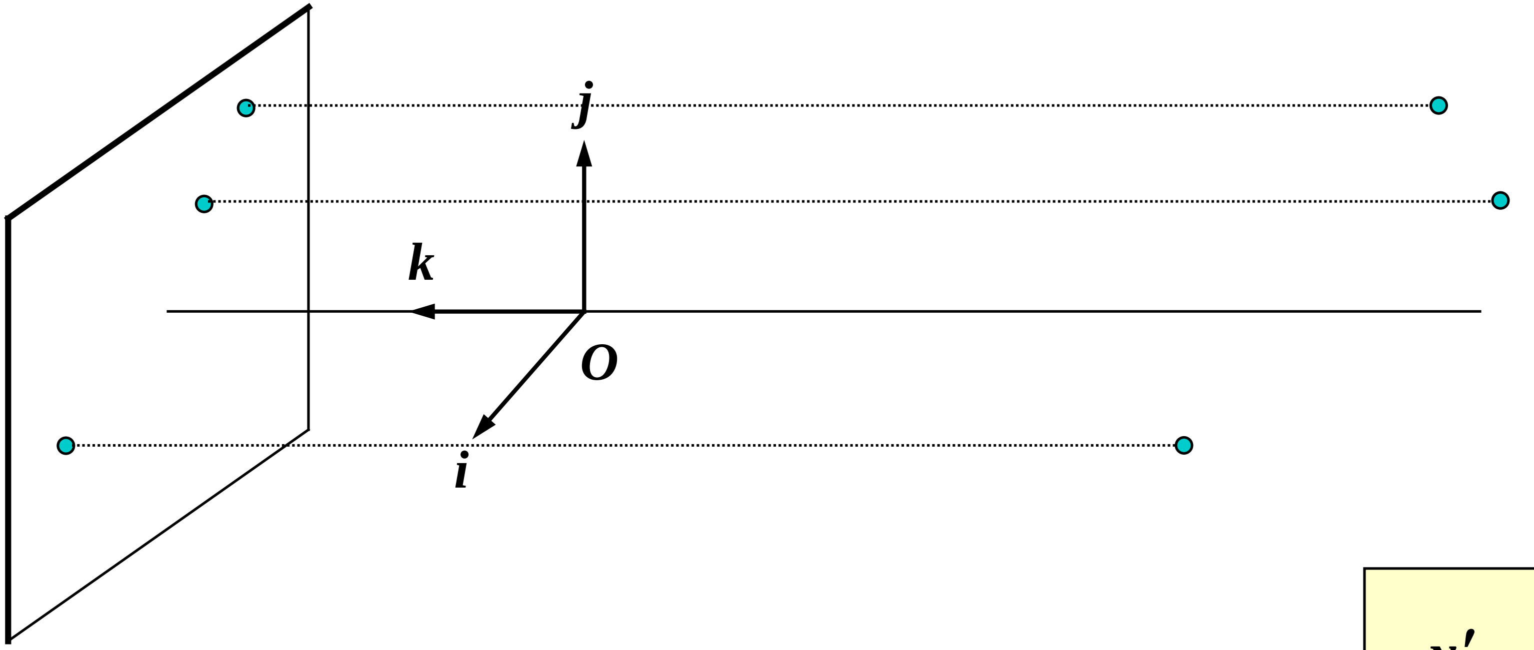
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

$$T_{31} = T_{32} = T_{33} = 0,$$

- x and y are linear combination of X, Y, Z
- No division is used to generate x and y
- There are many such Affine models

Orthographic projection

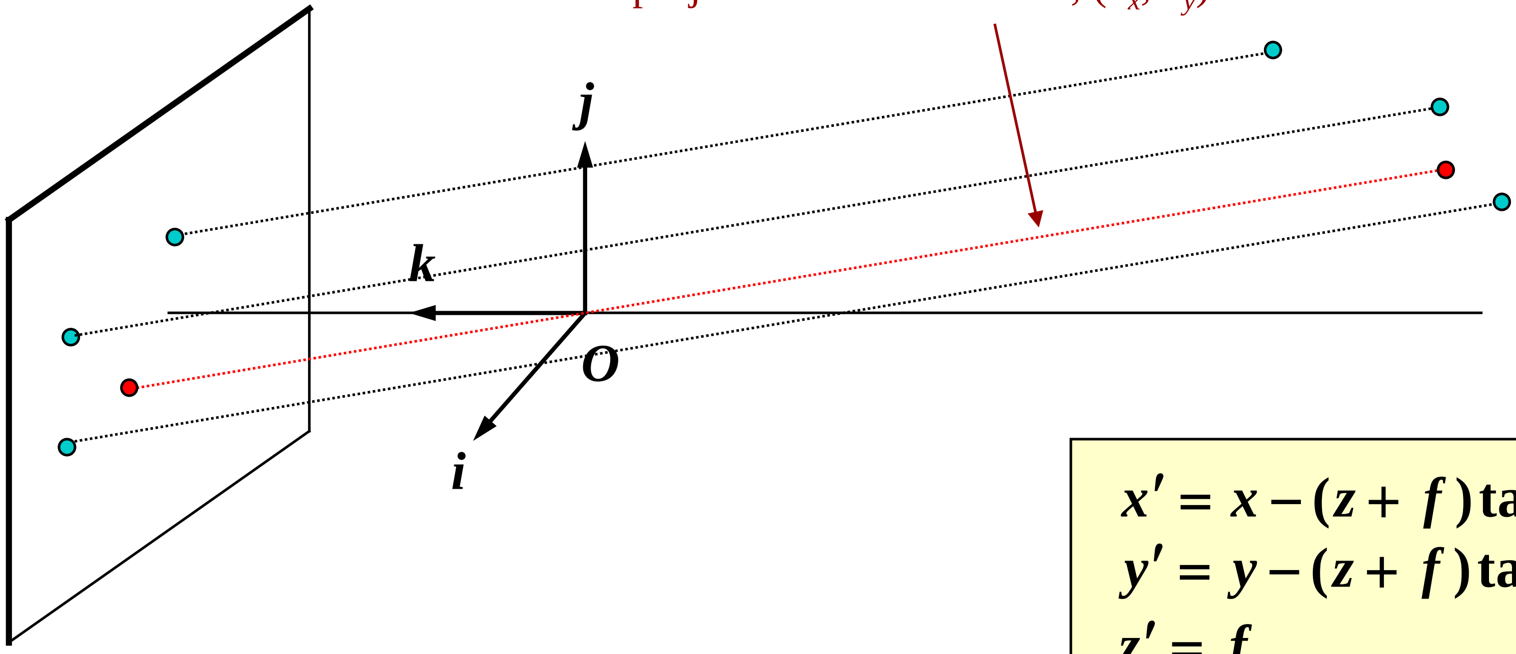
- All scene points are projected parallel to the optical axis (perpendicular to the image plane)



$$x' = x$$
$$y' = y$$
$$z' = f$$

Parallel projection

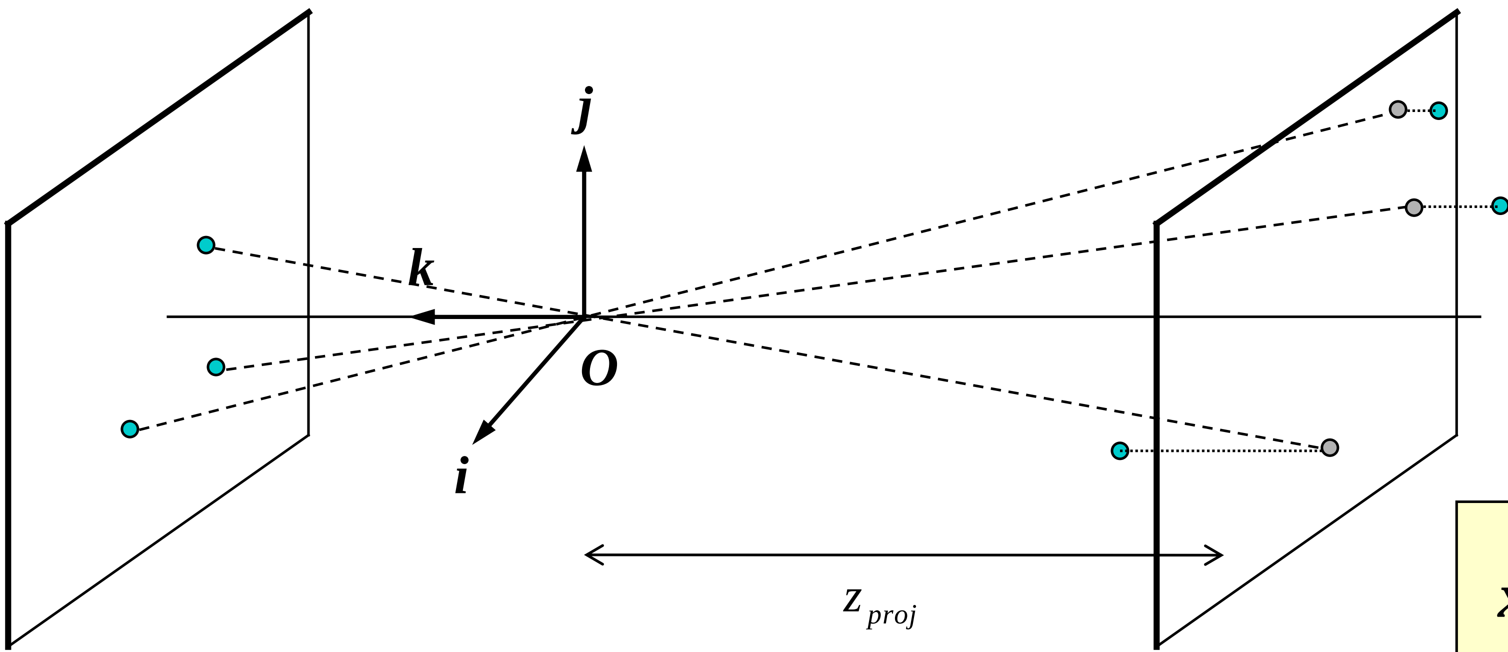
- All scene points are projected parallel to a reference direction (often defined by a central scene point)
  - Reference direction is not necessarily parallel to the optical axis
  - A generalization of orthographic projection



$$x' = x - (z + f)\tan\theta_x$$
$$y' = y - (z + f)\tan\theta_y$$
$$z' = f$$

Weak-perspective projection

- Scene points are orthographically projected onto a plane parallel to the image plane, then projected via perspective projection

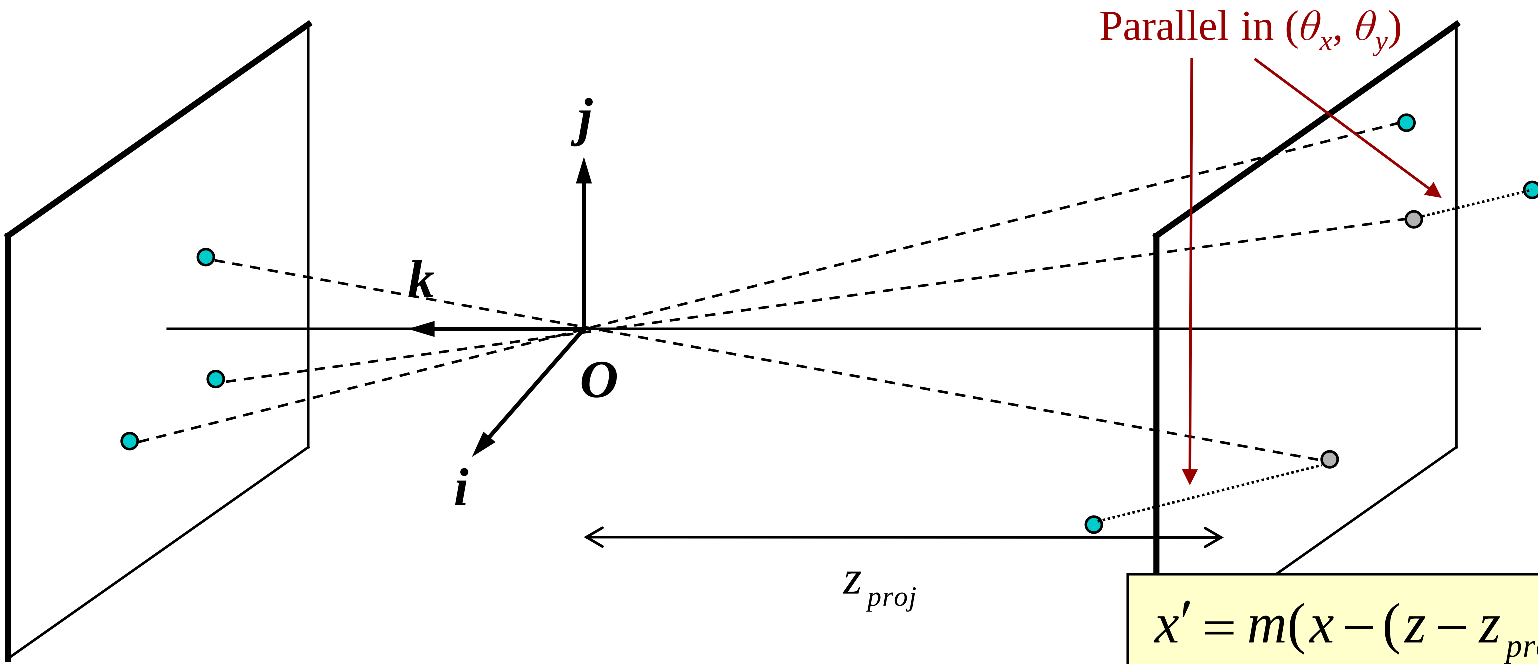


$$x' = mx$$
$$y' = my$$
$$z' = f$$

where $m = \dfrac{f}{z_{proj}}$

Paraperspective projection

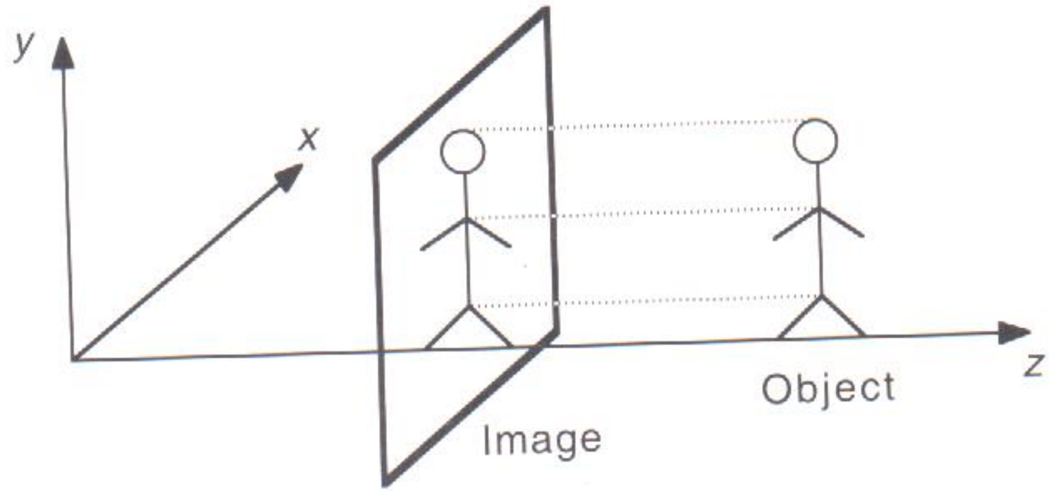
- Scene points are parallel projected onto a plane that is parallel to the image plane, then projected via perspective projection

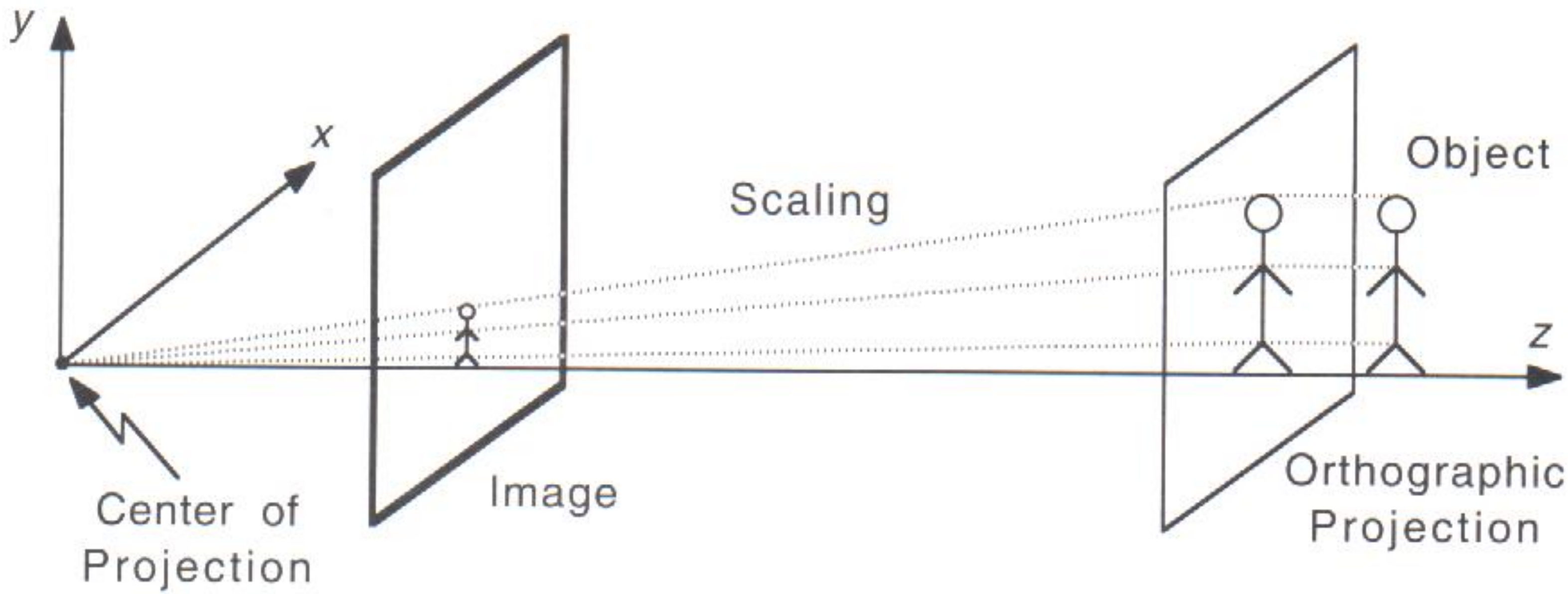


$$x' = m(x - (z - z_{proj})\tan\theta_x)$$
$$y' = m(y - (z - z_{proj})\tan\theta_y)$$
$$z' = f$$

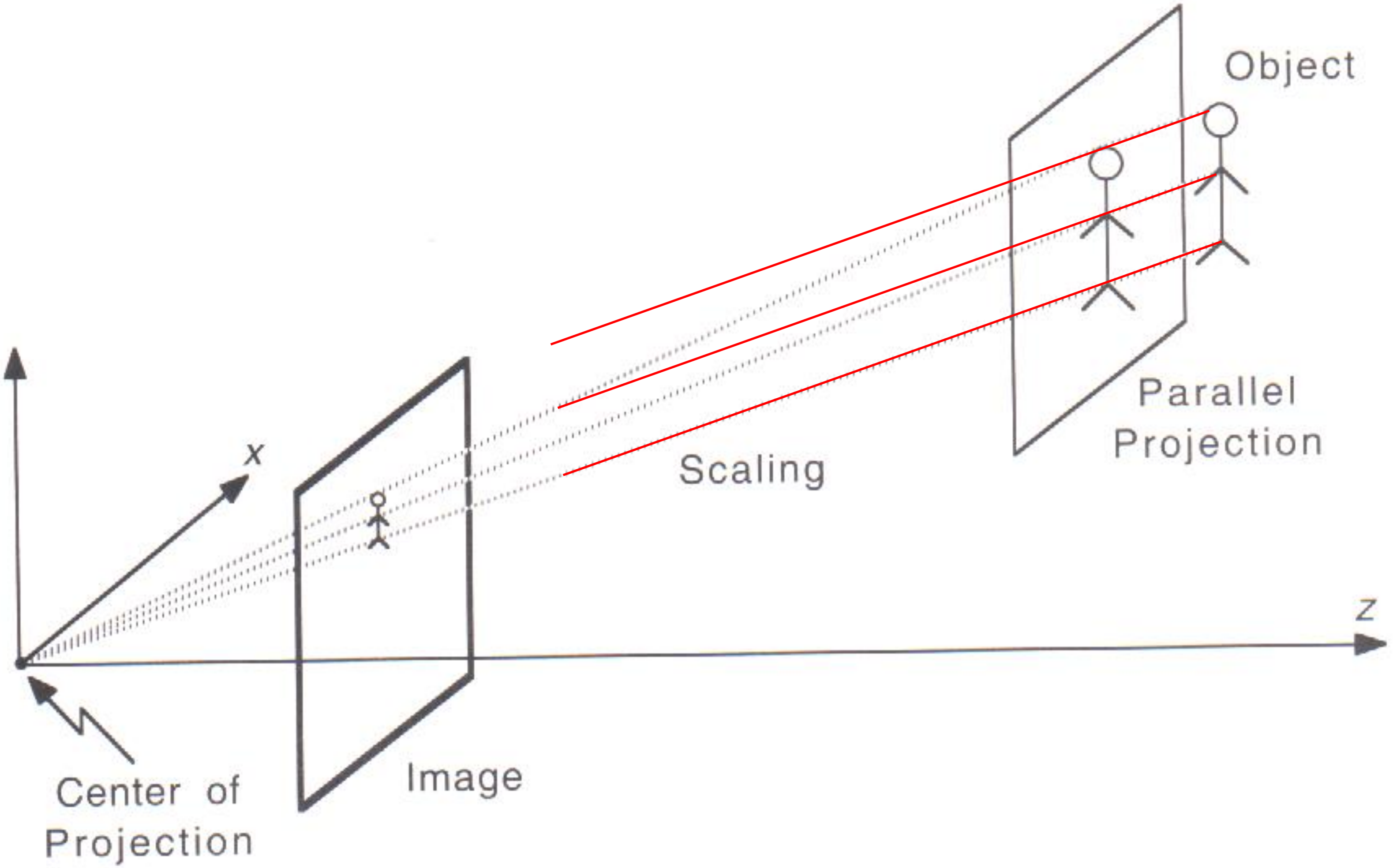
where $m = \dfrac{f}{z_{proj}}$

Orthographic and weak-perspective projection

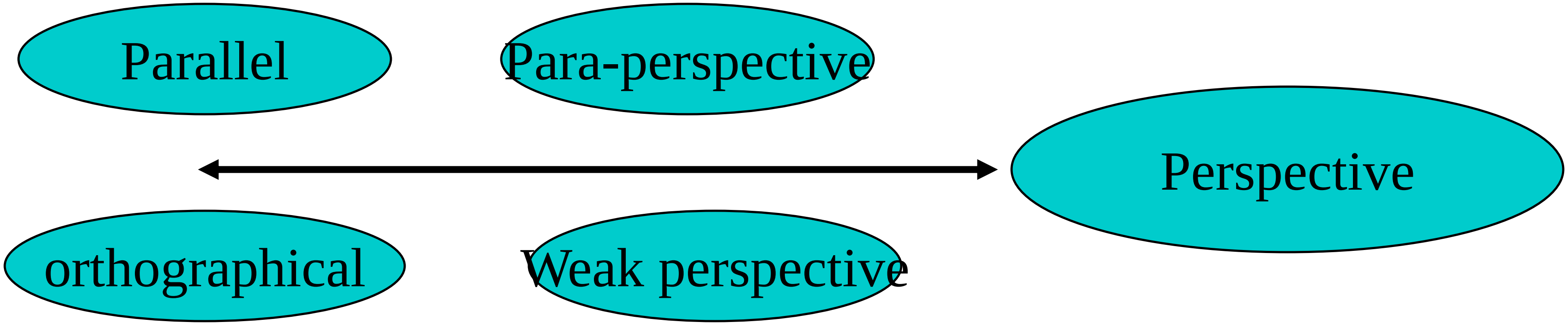




Paraperspective projection



Projection Models



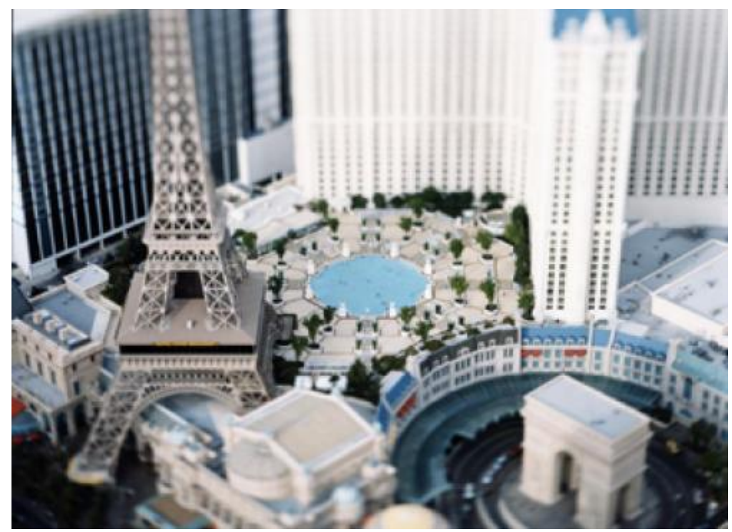
360 degree field of view…

- Basic approach
  - Take a photo of a parabolic mirror with an orthographic lens (Nayar)
  - Or buy one a lens from a variety of omnicam manufacturers…
    - See http://www.cis.upenn.edu/~kostas/omni.html

S. Seitz

Tilt-shift

http://www.northlight-images.co.uk/article_pages/tilt_and_shift_ts-e.html



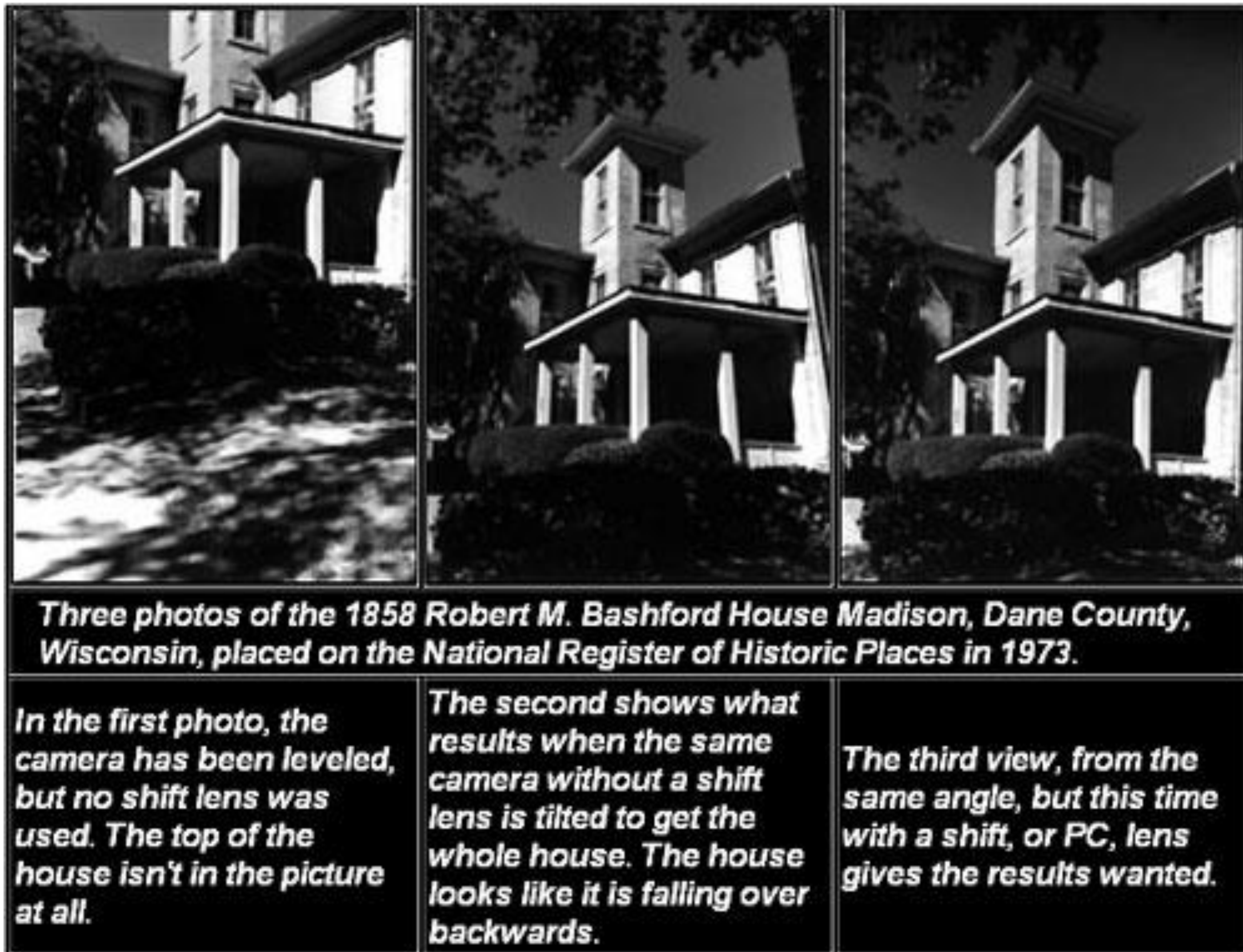
Titlt-shift images from Olivo Barbieri
and Photoshop imitations

S. Seitz

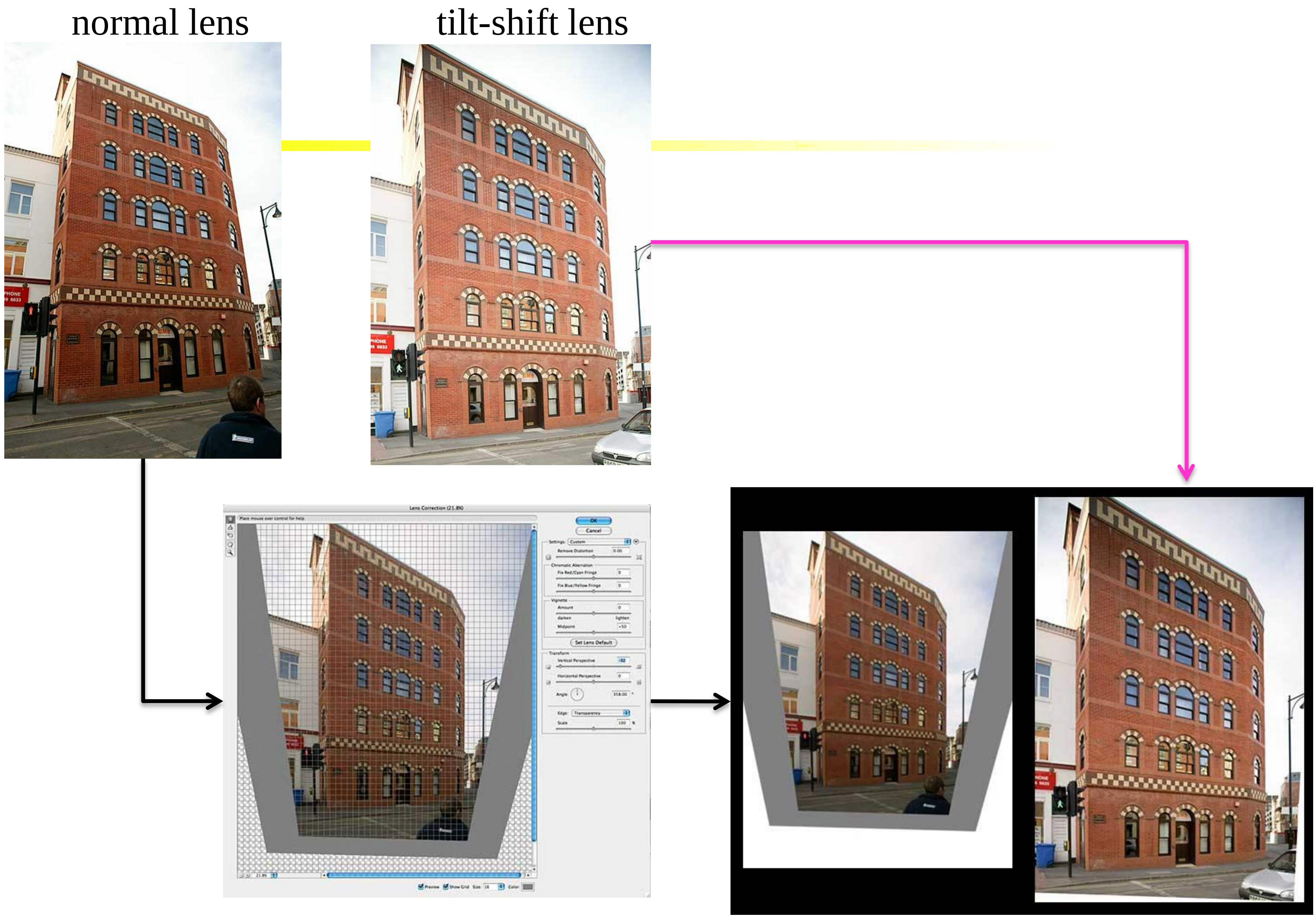
tilt, shift

http://en.wikipedia.org/wiki/Tilt-shift_photography

Tilt-shift perspective correction

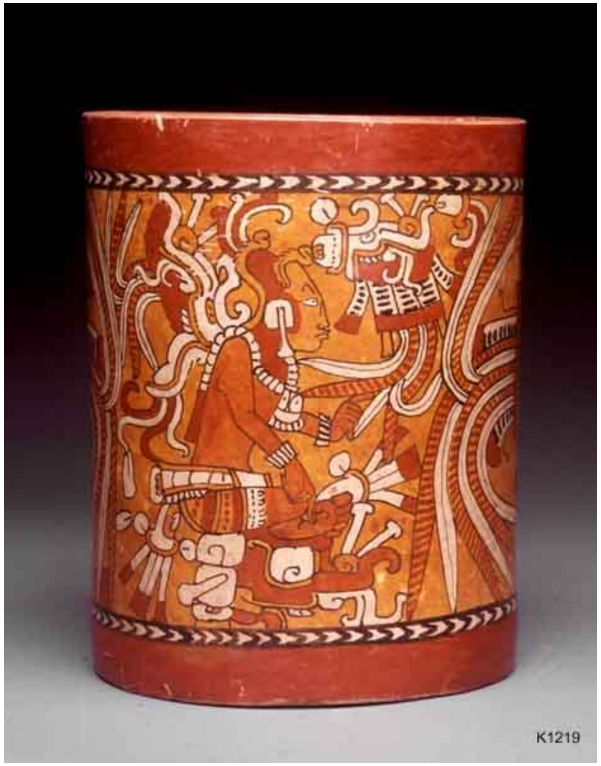


http://en.wikipedia.org/wiki/Tilt-shift_photography



http://www.northlight-images.co.uk/article_pages/tilt_and_shift_ts-e.html

Rotating sensor (or object)



Rollout Photographs © Justin Kerr
http://research.famsi.org/kerrmaya.html

Also known as "cyclographs", "peripheral images"

S. Seitz

Summary

- Two variations
  - Direction: parallel rays but do not have to be parallel to the optical axis
  - “Average” distance: object is roughly planar on a plane that is parallel to the image plane. Division by z is not necessary
- Why?
  - Linearity
- Examples
  - Carlo Tomasi and Takeo Kanade. (November 1992.). "Shape and motion from image streams under orthography: a factorization method.". *International Journal of Computer Vision*, **9** (2): 137–154

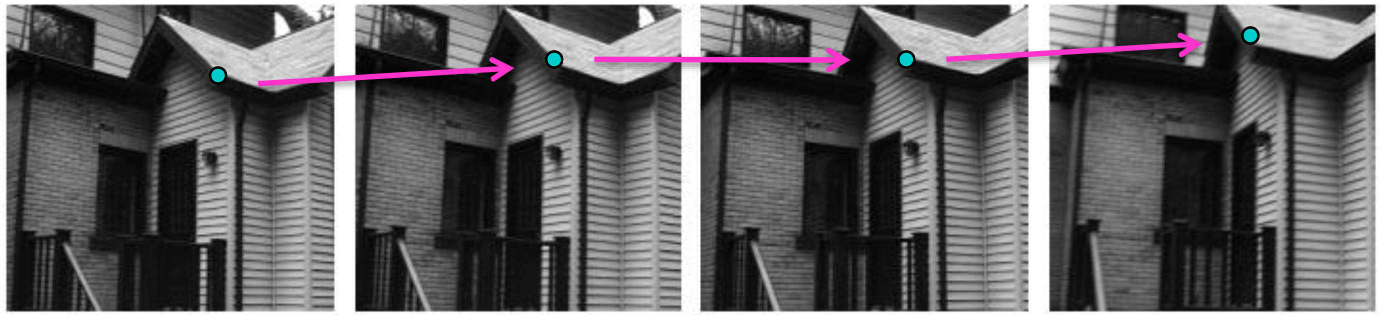
Factorization Method

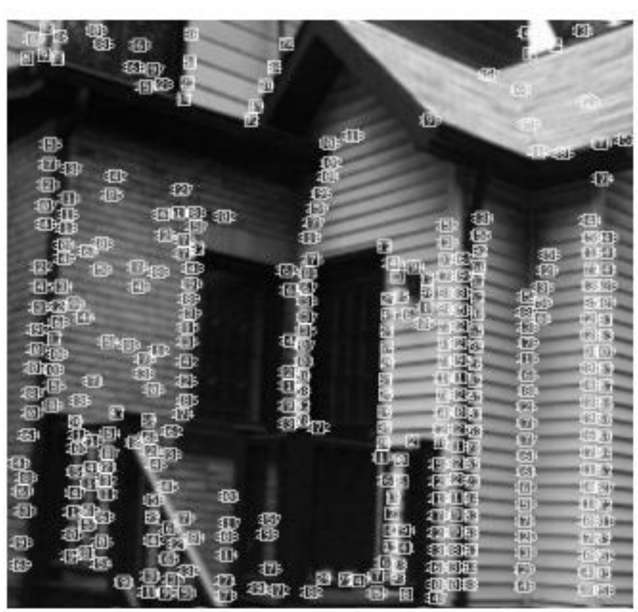
$$\mathbf{P} = \begin{pmatrix} x_{11} & \cdots & x_{1P} \\ y_{11} & \cdots & y_{1P} \\ \vdots & \ldots & \vdots \\ x_{F1} & \cdots & x_{FP} \\ y_{F1} & \cdots & y_{FP} \end{pmatrix}$$

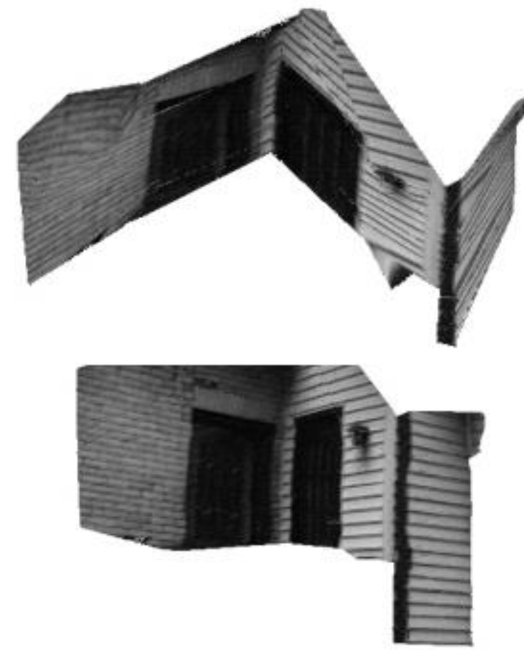
$$\mathbf{P} = \mathbf{MS}.$$

M: camera poses

S: object structures

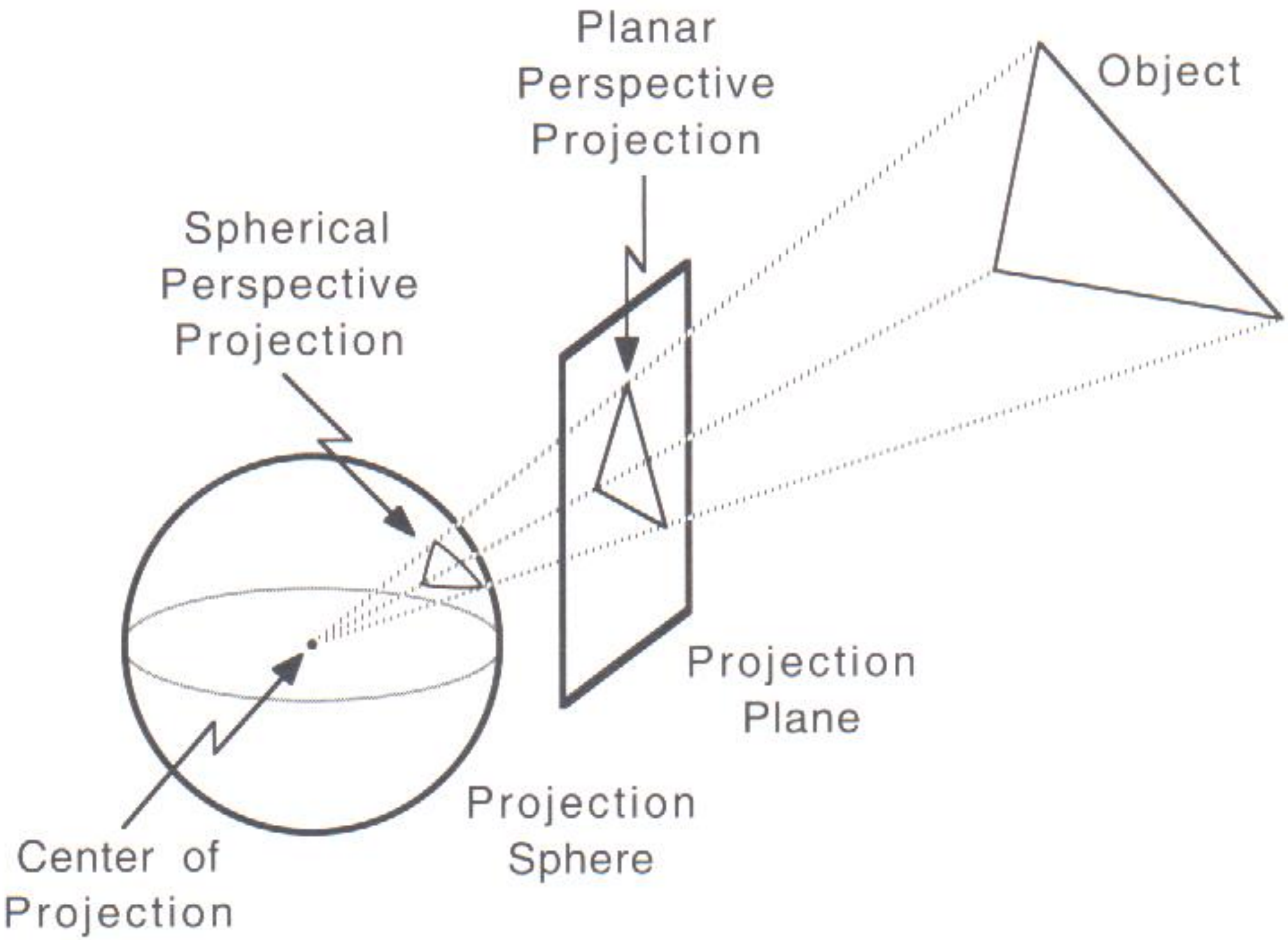






Reprinted from Tomasi and Kanade 1992

Planar and spherical projection

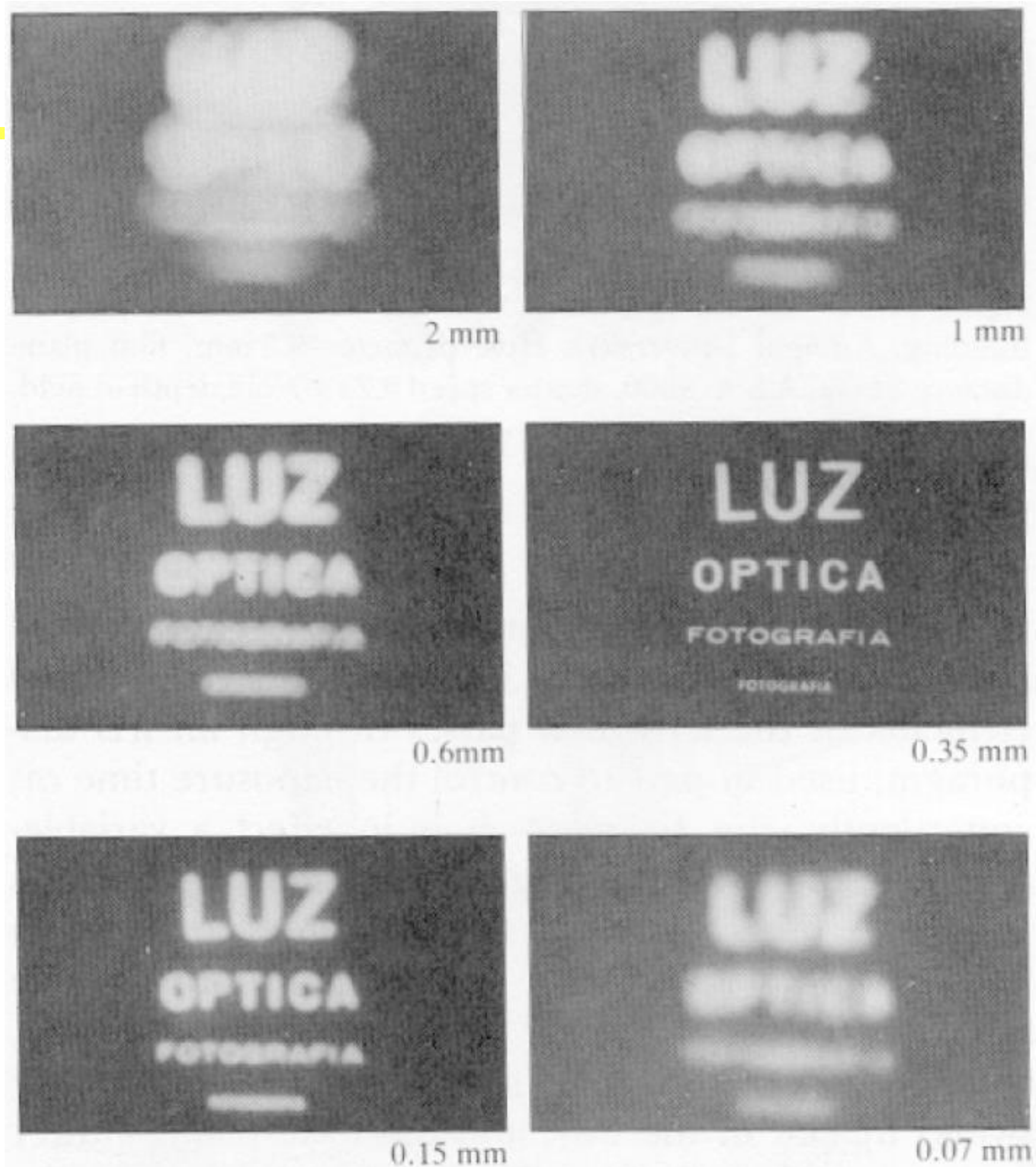


What do straight lines map onto under spherical perspective projection?

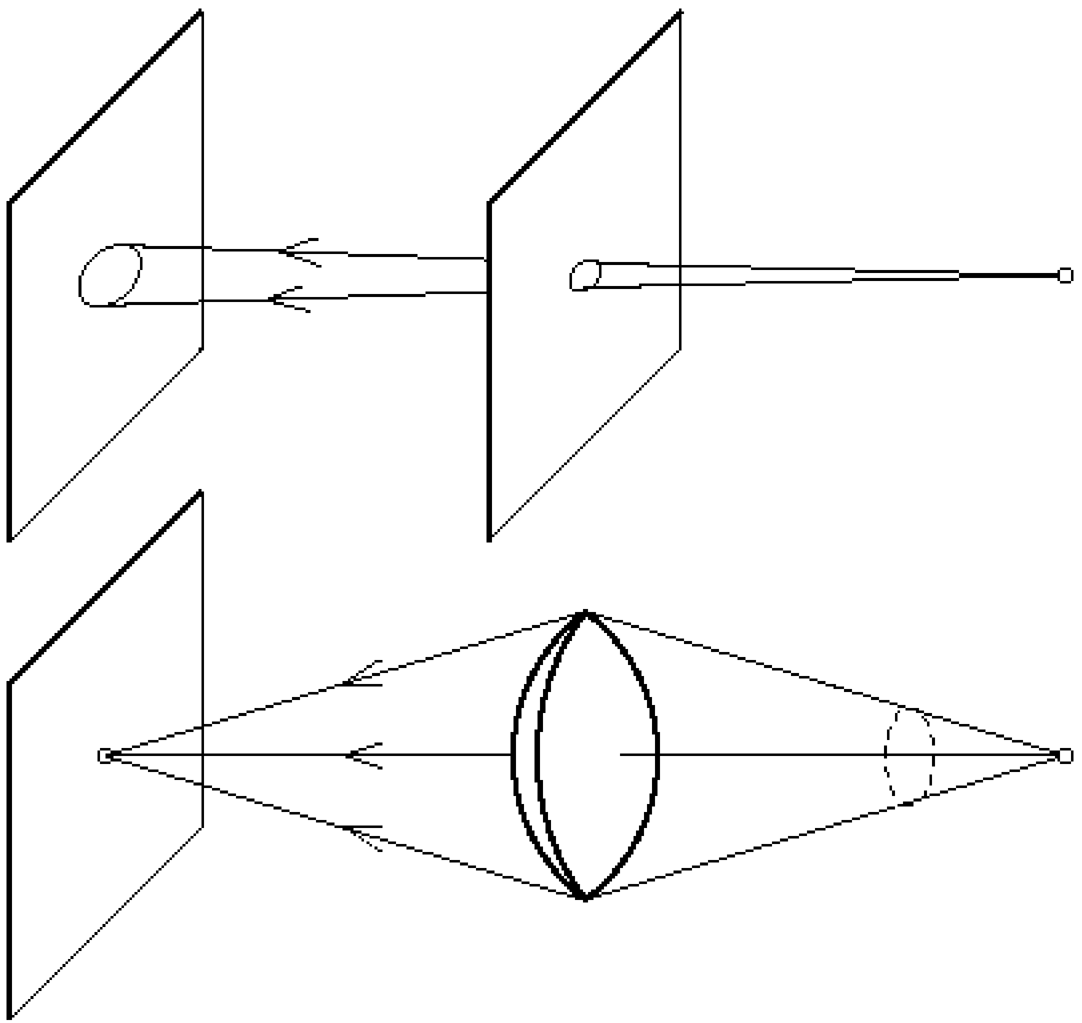
Pinhole too big - many directions are averaged, blurring the image

Pinhole too small- diffraction effects blur the image

Generally, pinhole cameras are *dark*, because a very small set of rays from a particular point hits the screen.

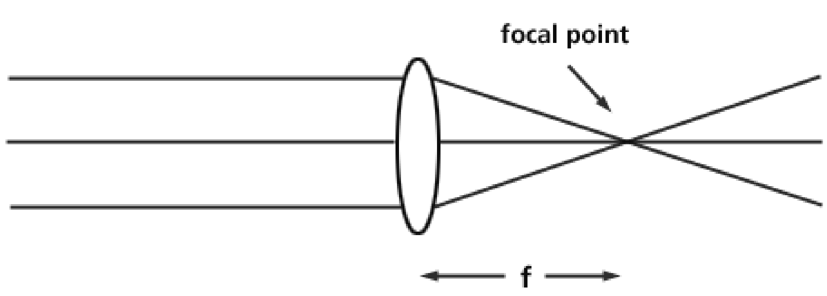


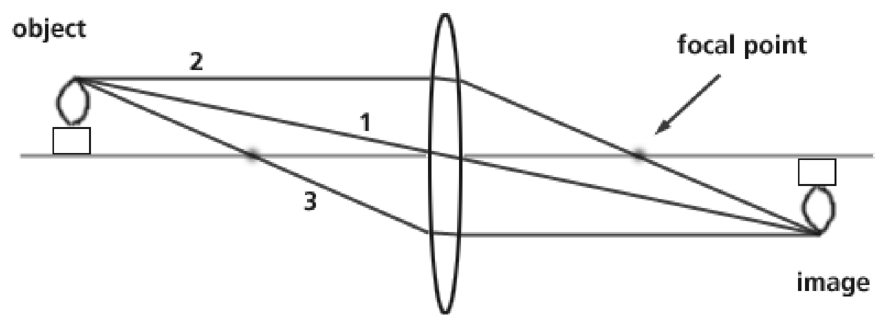
The reason for lenses



Optical Image Formation

- Image formation is achieved with a lens system integrated with optical elements which alter light, such as prisms and mirrors
- Lens – a transparent element, usually made of glass
  - Surfaces are usually spherically ground and polished
  - Refraction at surfaces "bends" the light rays and focuses them
  - The refracting power of a lens is related to its ***focal point***



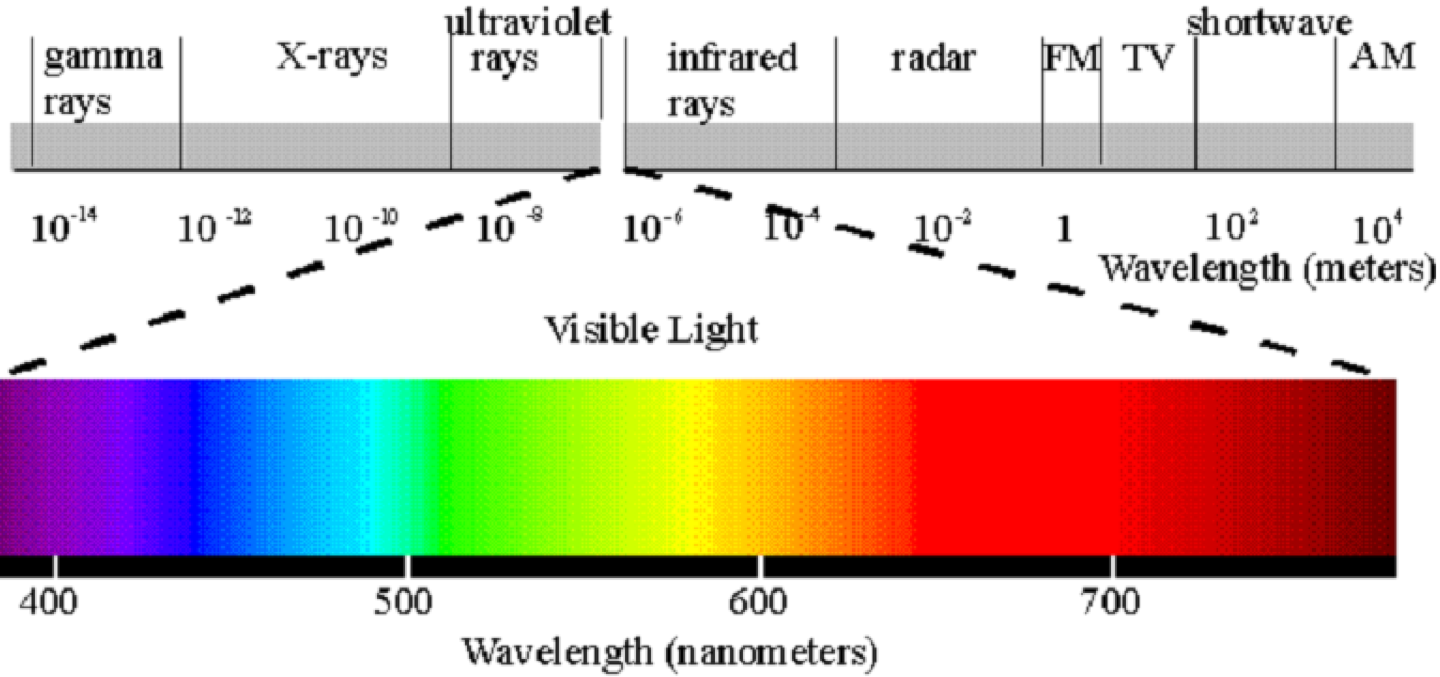


Light and Optics

- Optics – The branch of physics that deals with light
  - Ray optics
  - Wave optics
  - Photon optics
- Light – The visible portion of the electromagnetic spectrum; visible radiant energy
  - We're also interested in other parts of the EM spectrum (e.g., X-rays, UV rays, infrared)

University of California Santa Barbara

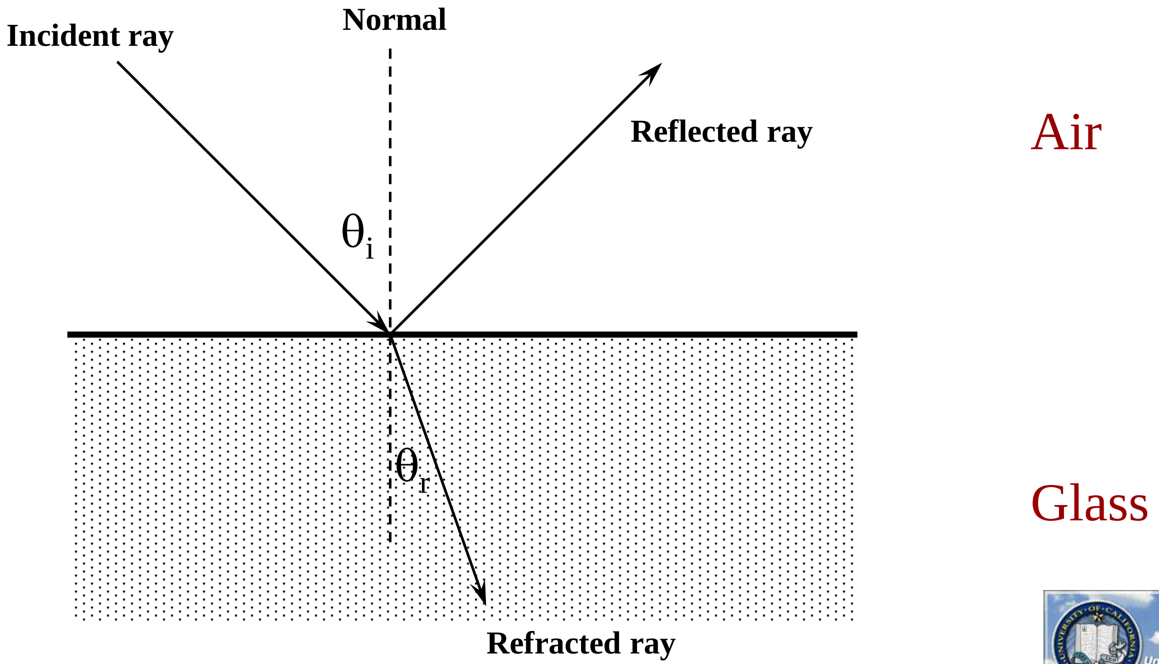
Electromagnetic (EM) Spectrum



Energy, frequency, and wavelength
are related

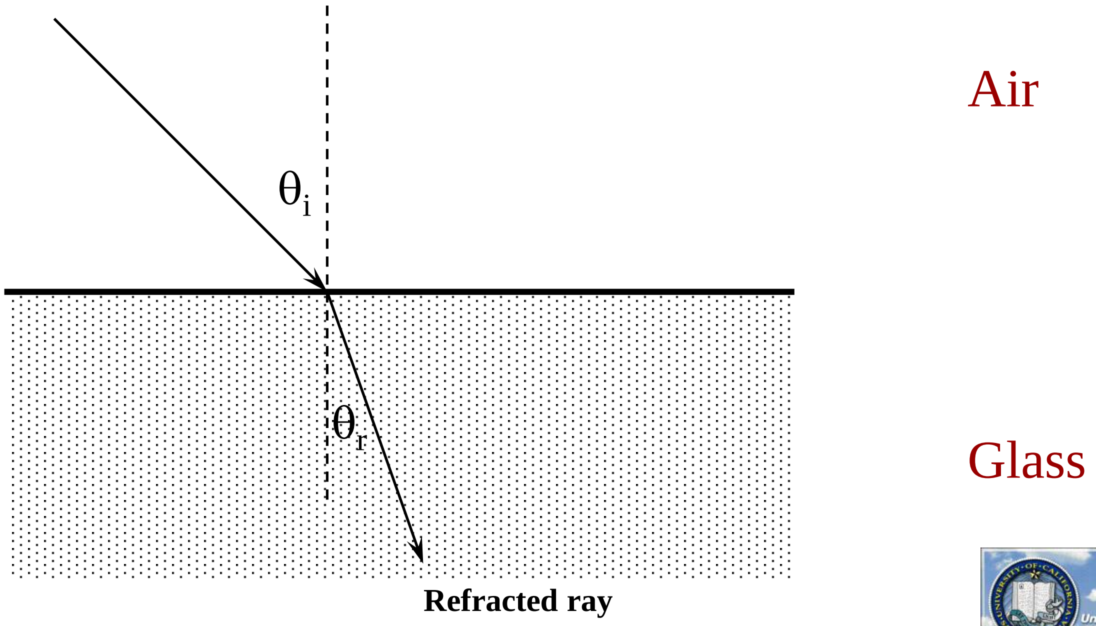
Reflection and Refraction

- At the interface between media, there is reflection and refraction (and absorption)
  - **Reflection** – light is reflected from the surface
  - **Refraction** – light proceeds through the material in a different direction, depending on the index of refraction

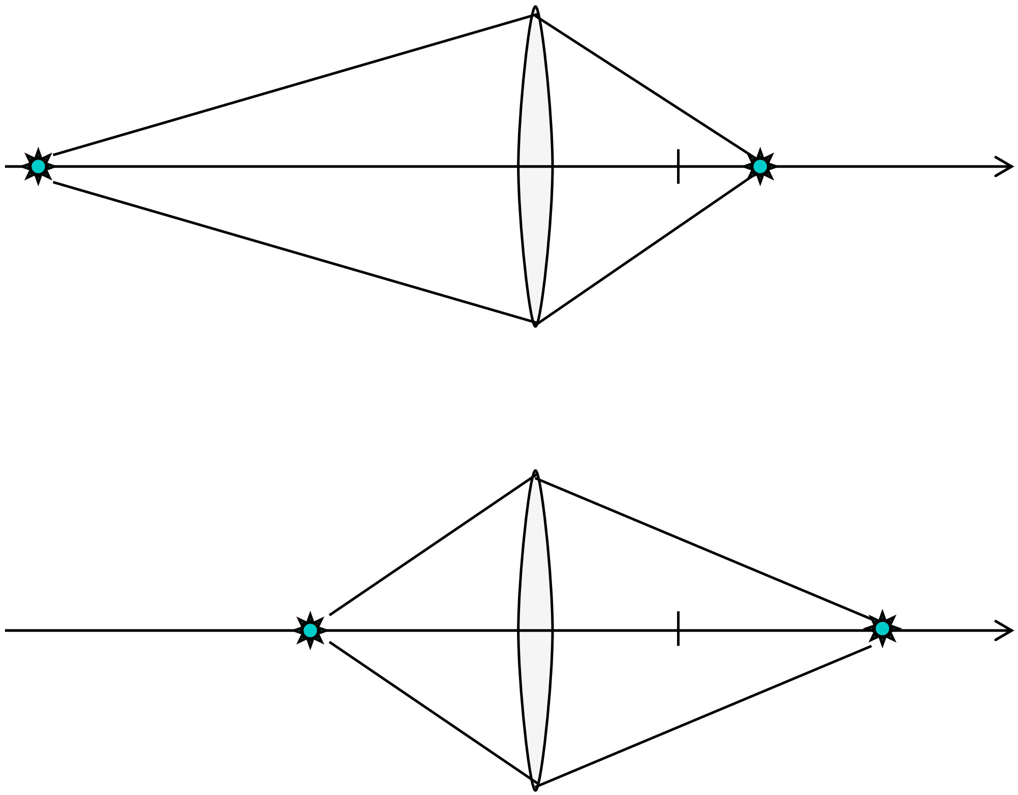


Refraction

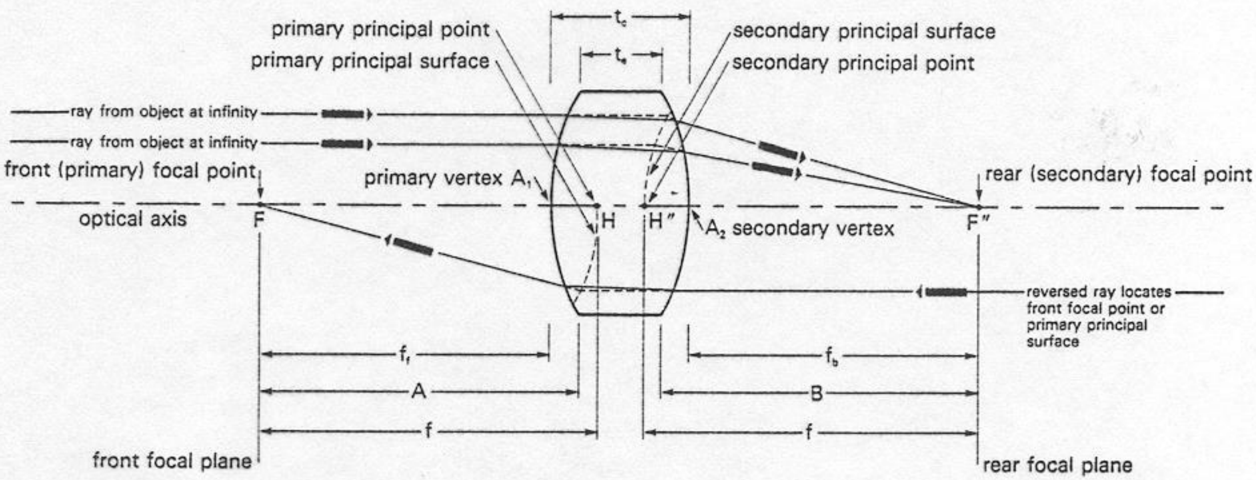
- Refraction occurs toward the normal when light enters a more dense medium
- Medium characterized by its *index of refraction* (*n*)
  - Index of refraction defined as $\boldsymbol{n = c / v_m}$
- Snell's Law for refraction: $n_1 \sin(\theta_i) = n_2 \sin(\theta_r)$



Lens refraction causes points to focus



Thick Lens



f = Effective focal length (EFL), may be positive (as shown) or negative

f_f = Front focal length (FFL)

f_b = Back focal length (BFL)

A = Front focus to front edge distance

B = Rear edge to rear focus distance

t_e = Edge thickness

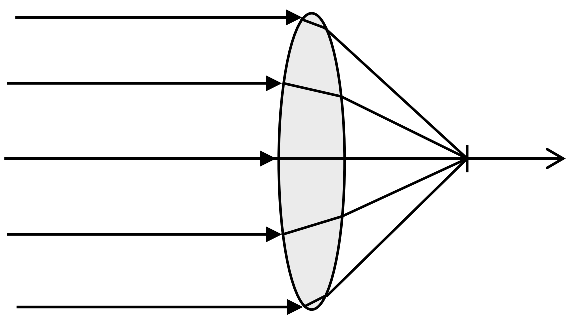
t_c = Center thickness

FRONT AND BACK FOCAL LENGTHS of a lens having spherical surfaces and surrounded by air. Under these conditions, distances labeled f are equal whether or not the lens is symmetric, but distances f_f and f_b are equal *only* if the lens is symmetric. In the paraxial limit (see text), the curvature of the principal surfaces may be neglected.

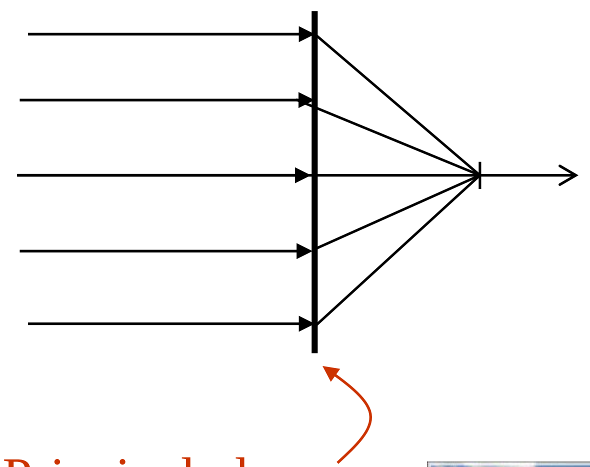
Thin Lens

- The thin lens model is an ideal approximation to a lens
  - Disregard the lens thickness
  - Disregard blurring (rays don't really converge)
  - All light bending takes place at the principal plane, where the incoming rays intersect with the outgoing rays

Real lens



Ideal lens



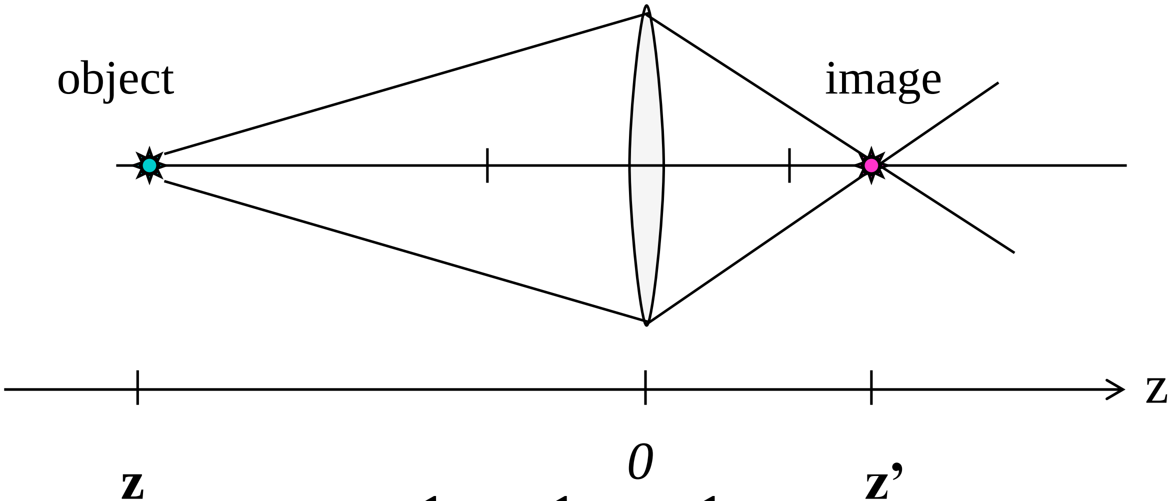
Principal plane

Focal Point

$$\frac{1}{f} = (n-1)\left(\frac{1}{r_1} + \frac{1}{r_2}\right)$$

Focal distance is a function of the index of refraction (n) and the surface radii for the two sides of the lens

Thin Lens Equation



$$\frac{1}{z} + \frac{1}{z'} = \frac{1}{f}$$

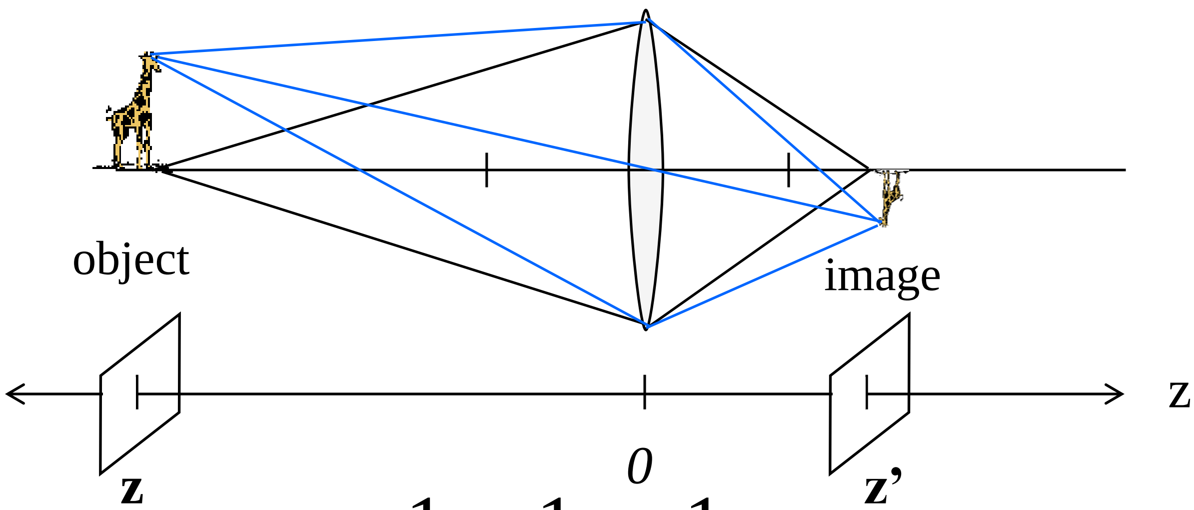
(Careful about signs!)

z – distance of point/object from lens origin (positive outside)

z' – distance from lens origin where point/object is in focus (positive inside)

f – focal distance (constant property of the lens, positive convex, negative concave)

Focus



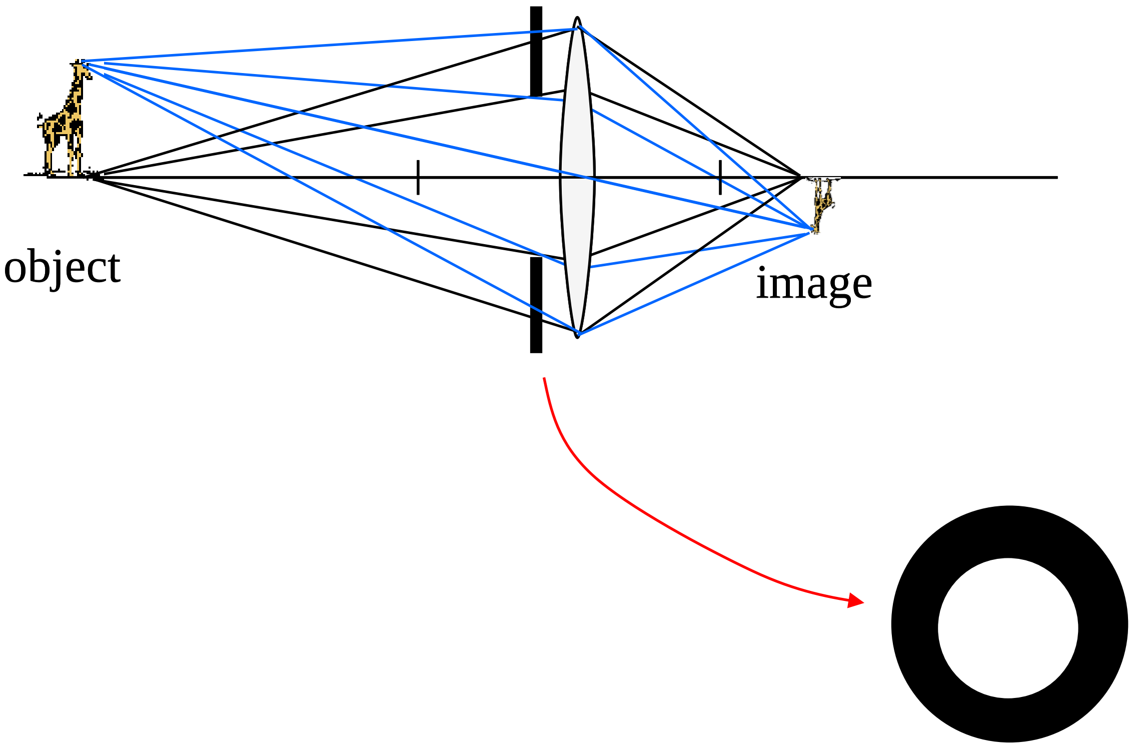
$$\frac{1}{z} + \frac{1}{z'} = \frac{1}{f}$$

So everything in the plane Z=z will be imaged (focused) at Z=z'

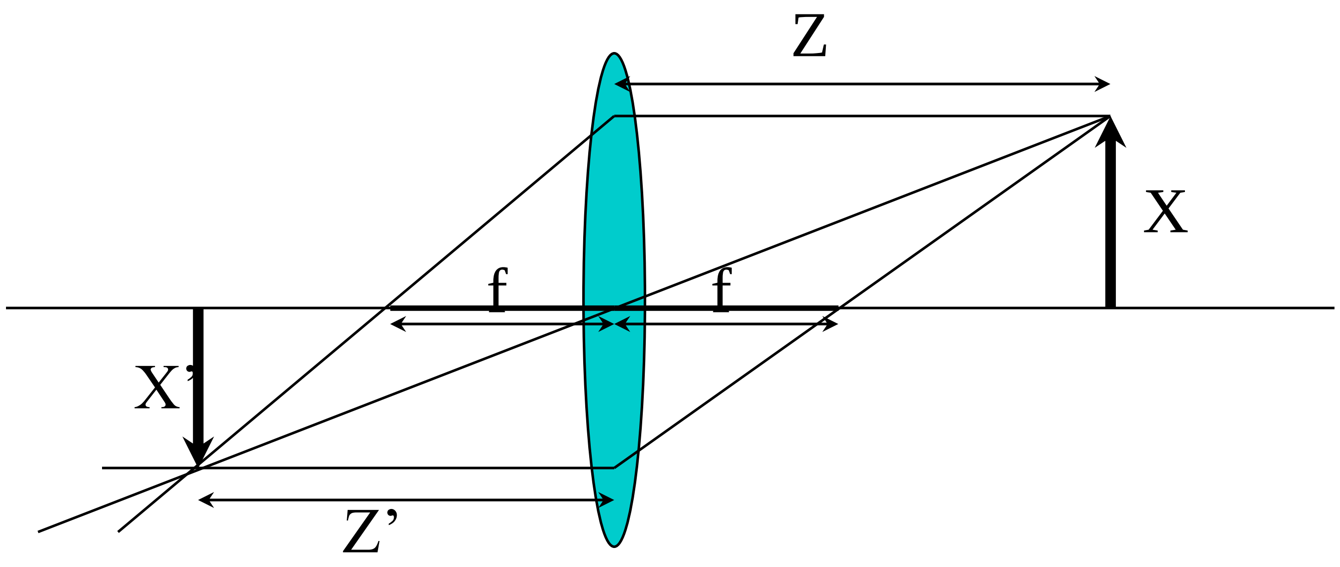
Everything else will be out of focus (sort of)

Aperture

- An aperture limits the area that light can pass through



Relation to Pin Hole Model



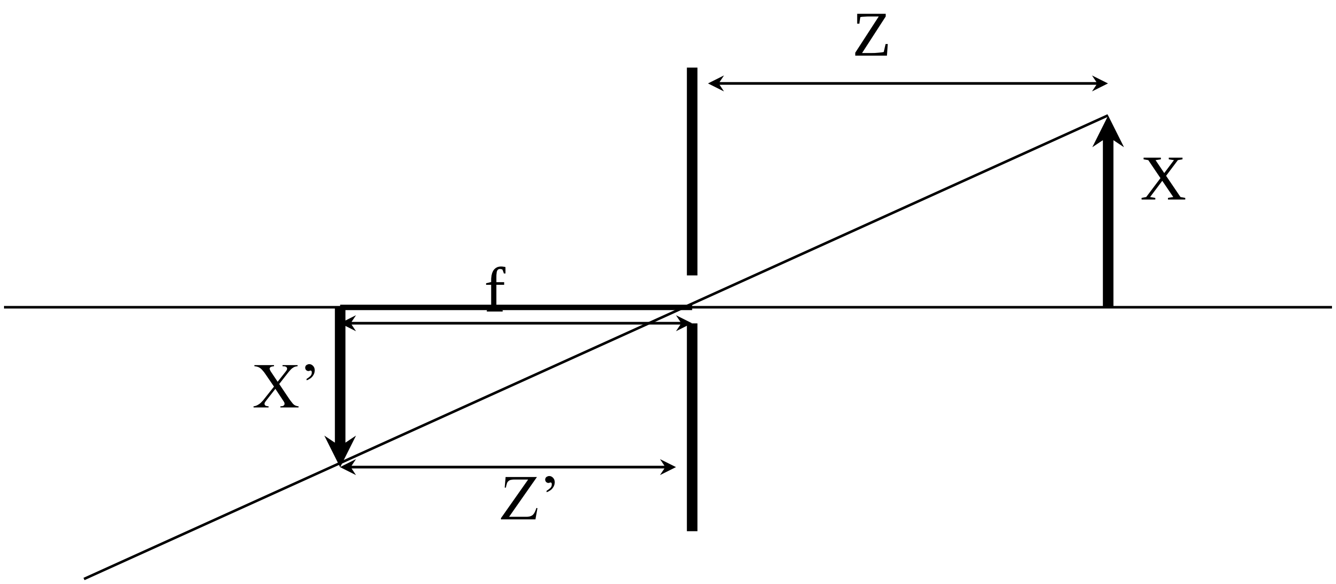
$$\frac{1}{Z} + \frac{1}{Z'} = \frac{1}{f}$$

$$X' = -Z' \frac{X}{Z}$$

Image Formation

- In general
- A pin-hole model

$$Z >> f, \frac{1}{Z} << \frac{1}{f} \Rightarrow \frac{1}{Z'} \approx \frac{1}{f} \Rightarrow Z' \approx f$$

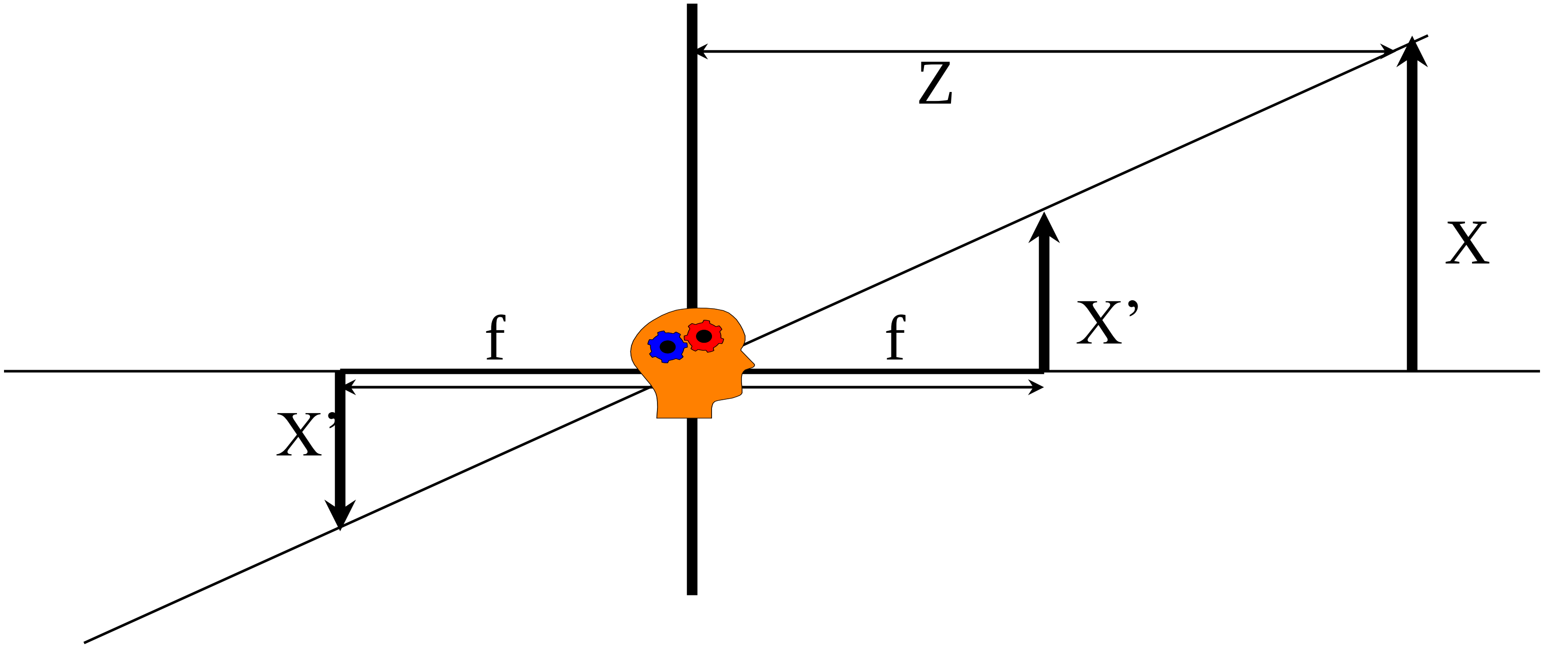


$$X' = -f\frac{X}{Z}$$

$$Y' = -f\frac{Y}{Z}$$

Image Formation (cont.)

- A pin-hole model without inversion
- Back to old pinhole camera formula



$$X' = f\frac{X}{Z}$$

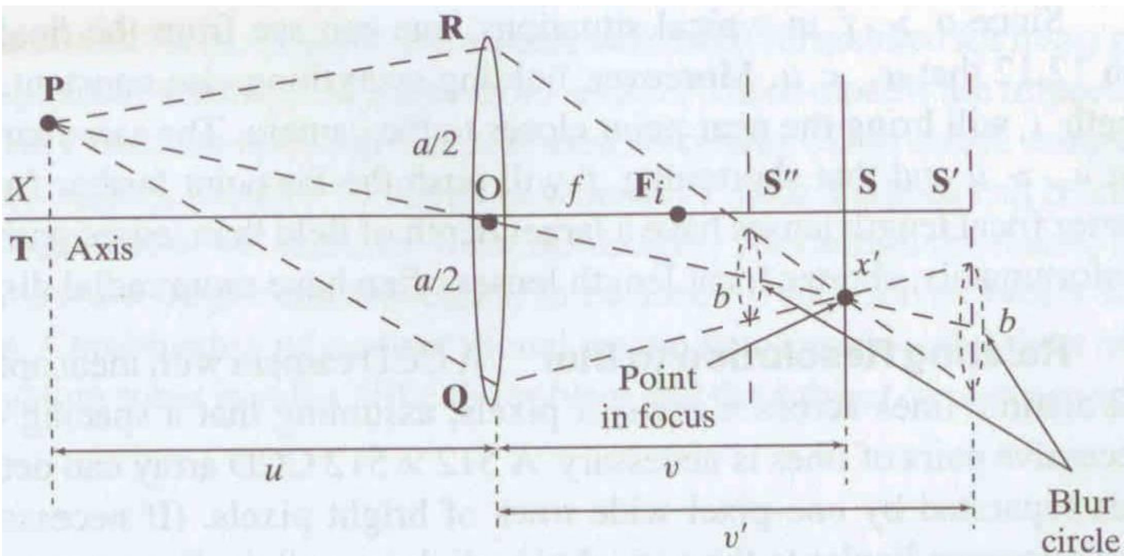
$$Y' = f\frac{Y}{Z}$$

Focus and depth of field

Image credit: cambridgeincolour.com

Focus and depth of field

- Depth of field: distance between image planes where blur is tolerable



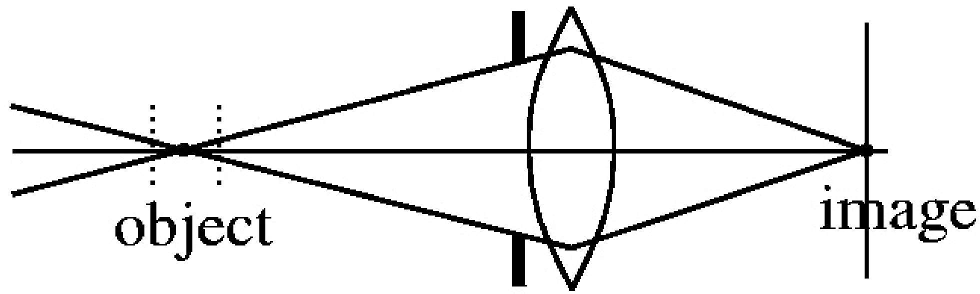
Thin lens: scene points at distinct depths come in focus at different image planes.

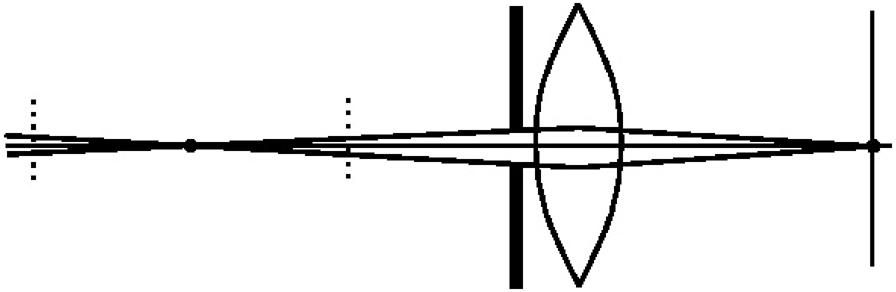
(Real camera lens systems have greater depth of field.)

← "circles of confusion" →

Focus and depth of field

❖ How does the aperture affect the depth of field?



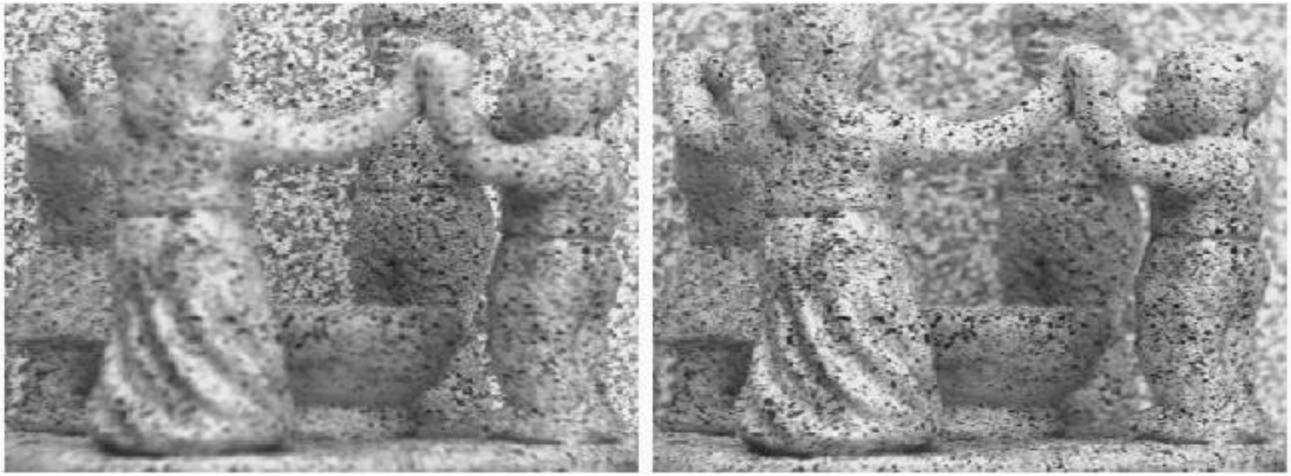


- A smaller aperture increases the range in which the object is approximately in focus

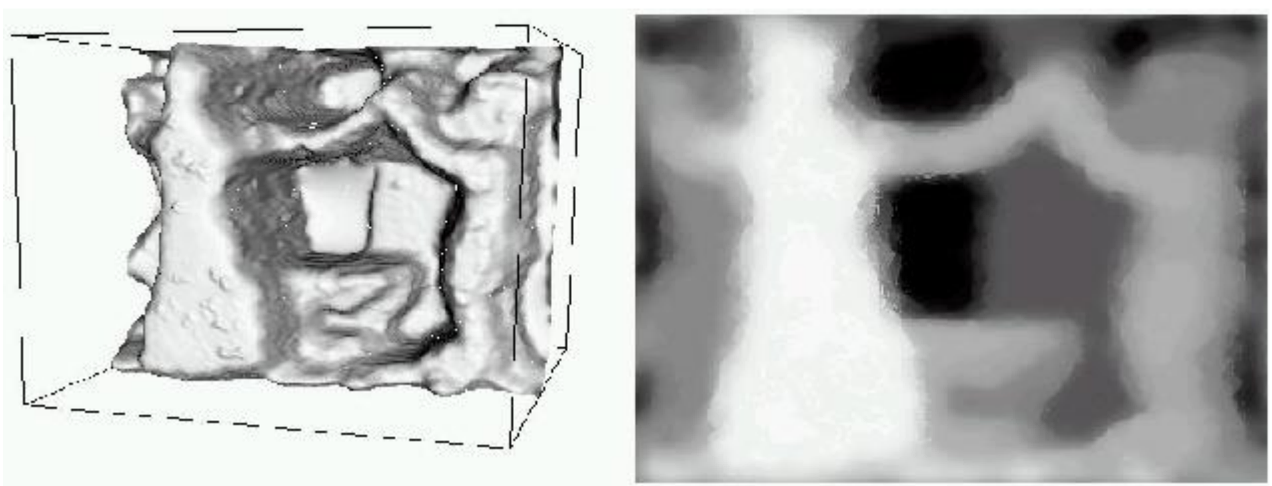
Flower images from Wikipedia http://en.wikipedia.org/wiki/Depth_of_field

Slide from S. Seitz

Depth from focus



Images from same point of view, different camera parameters



3d shape / depth estimates

Field of view

- Angular measure of portion of 3d space seen by the camera

28 mm lens, 65.5° × 46.4°

50 mm lens, 39.6° × 27.0°

70 mm lens, 28.9° × 19.5°

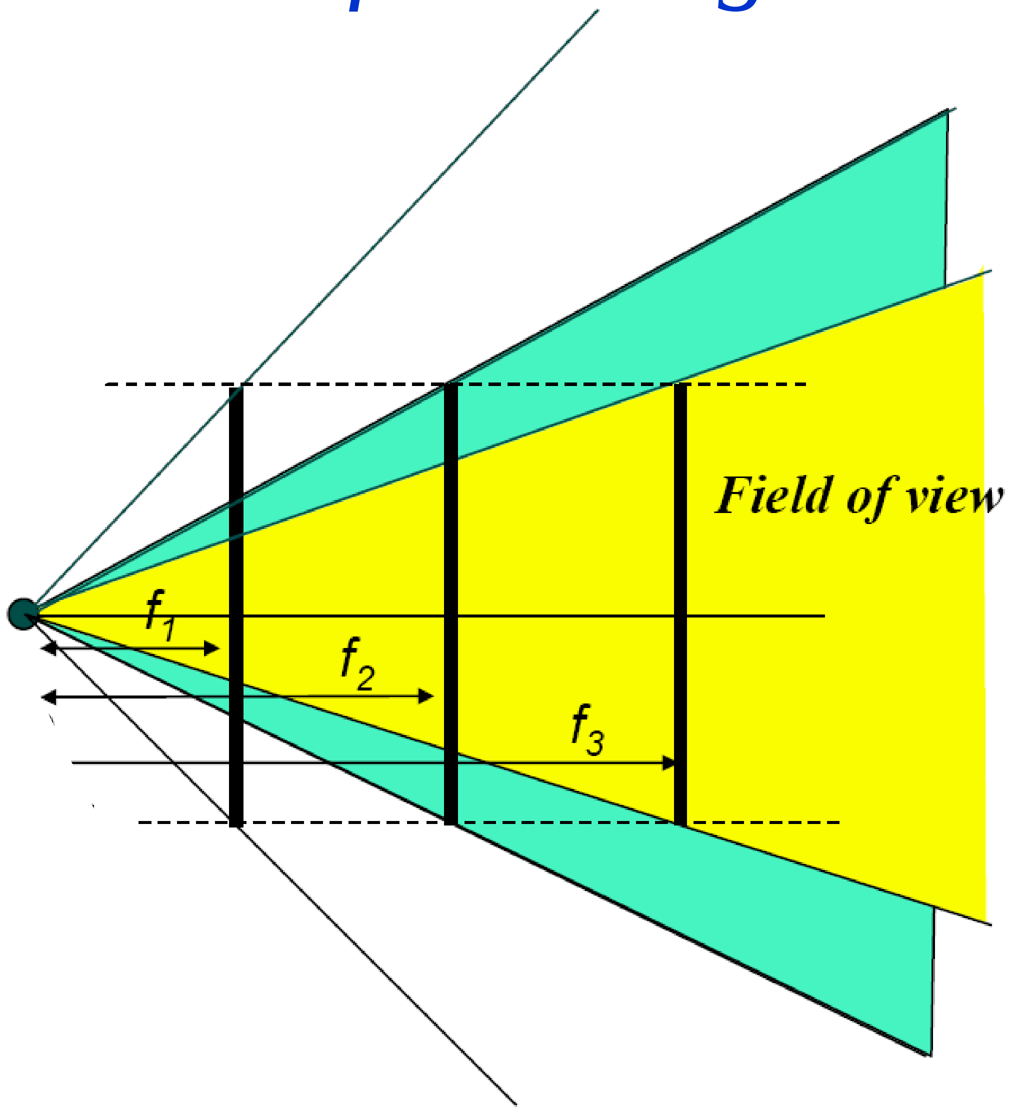
210 mm lens, 9.8° × 6.5°

Images from http://en.wikipedia.org/wiki/Angle_of_view

K. Grauman

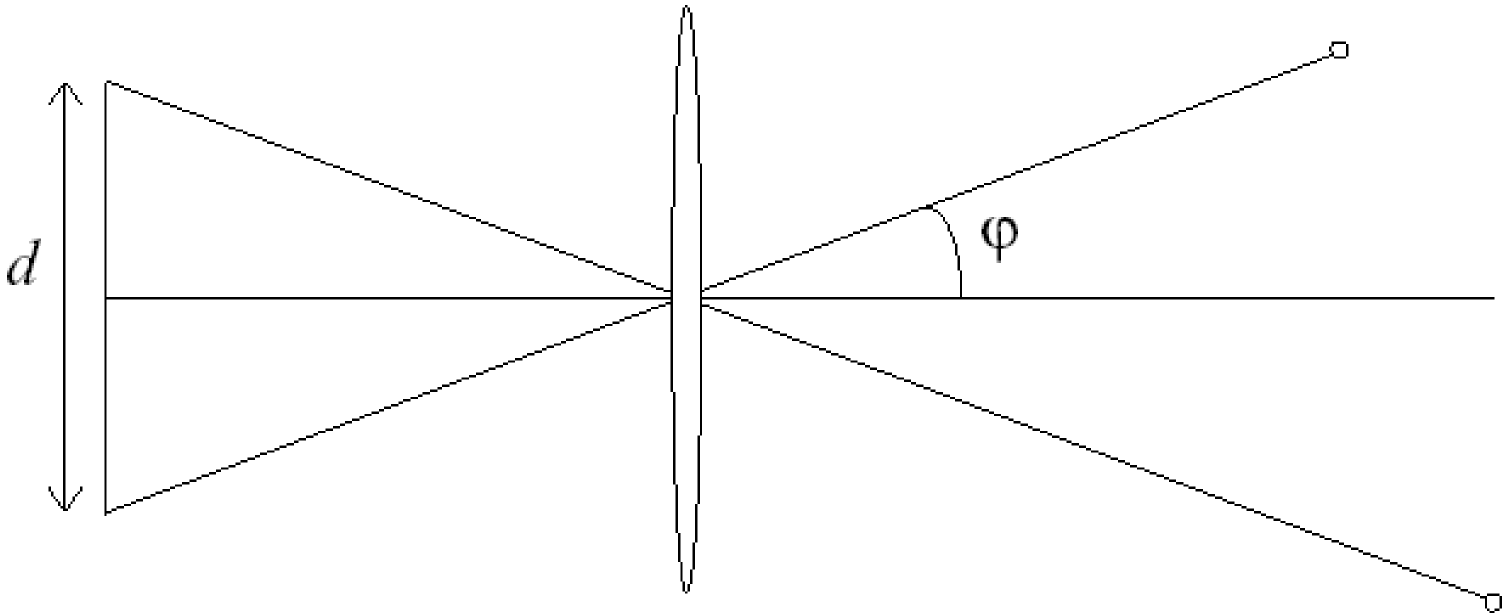
Field of view depends on focal length

- As ***f*** gets smaller, image becomes more *wide angle*
  - more world points project onto the finite image plane
- As ***f*** gets larger, image becomes more *telescopic*
  - smaller part of the world projects onto the finite image plane



from R. Duraiswami

Field of view depends on focal length



Size of field of view governed by size of the camera retina:

$$\varphi = \tan^{-1}\left(\frac{d}{2f}\right)$$

Smaller FOV = larger Focal Length

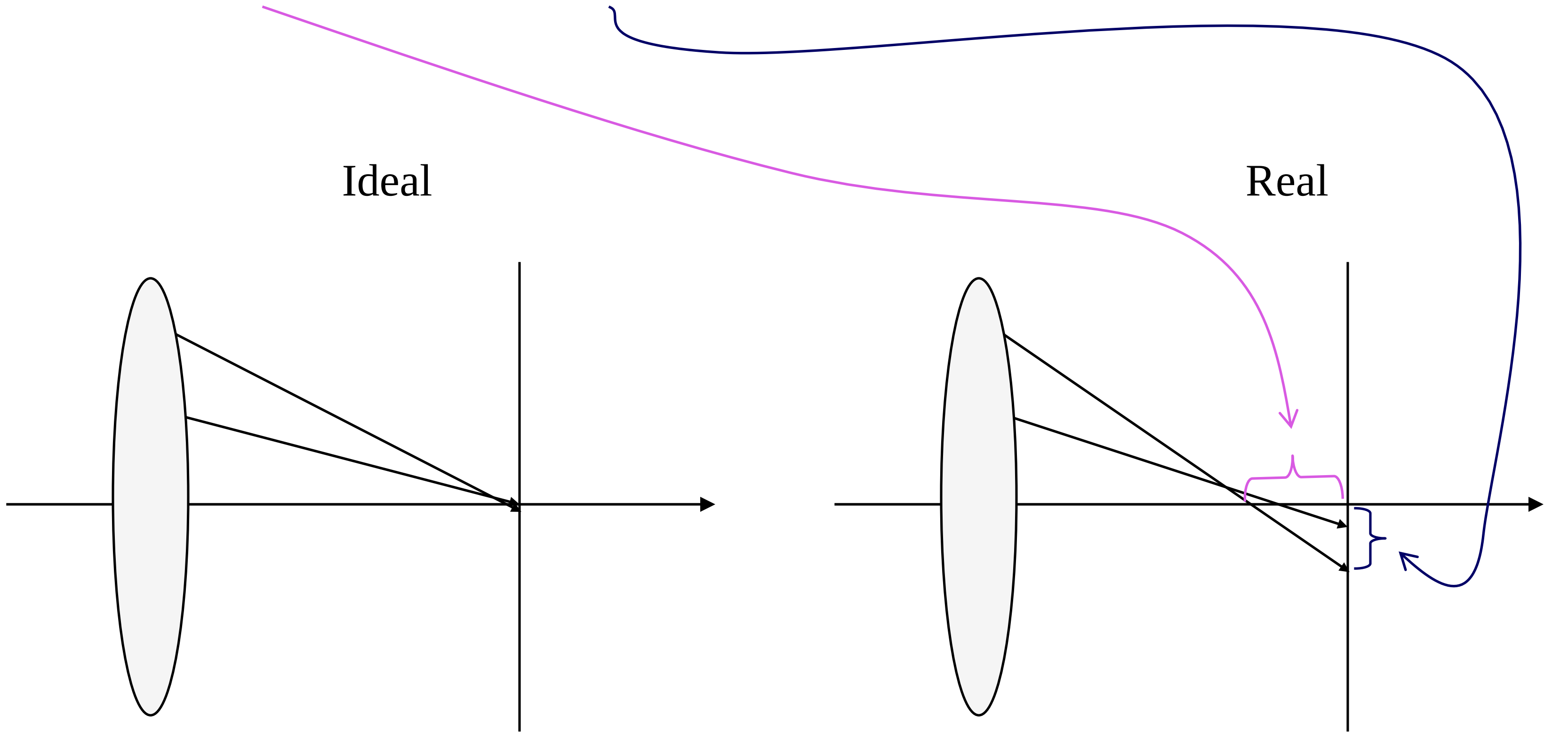
Slide by A. Efros

Lens Aberrations

- Aberrations: Systematic deviations from the ideal path of the image-forming rays
  - Causes blur and other problems in the image
  - Good optical design minimizes aberrations (but doesn’t eliminate them!)
  - Typically small effect for paraxial rays
- Types:
  - Spherical aberration
  - Coma
  - Astigmatism
  - Distortion
  - Chromatic aberration

Lens Aberrations

- Longitudinal and lateral (transverse) effects

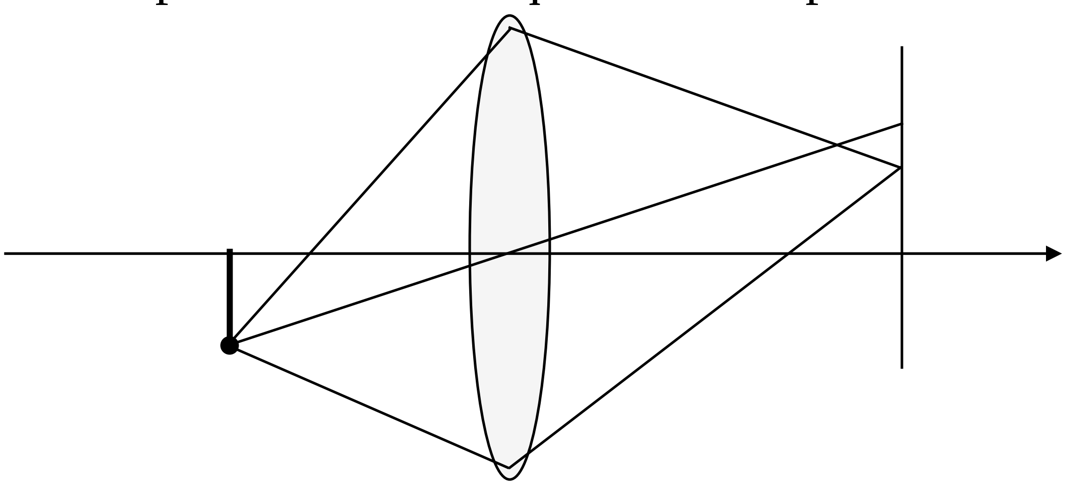


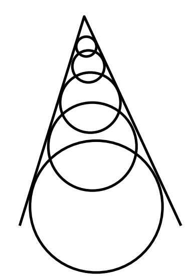
Spherical Aberration

- The spherical lens shape does not really effect perfect focusing. However:
  - It works reasonably well
  - It is by far the easiest to fabricate and measure
  - It is used in all but the most demanding situations
- Spherical aberration causes different focal lengths for light rays entering at different parts of the lens (varies with radial distance)
  - a longitudinal aberration
- Effects can be reduced by using two lenses together, convex and concave
- Blurs the image
- Proportional to the diameter of the lens

Coma

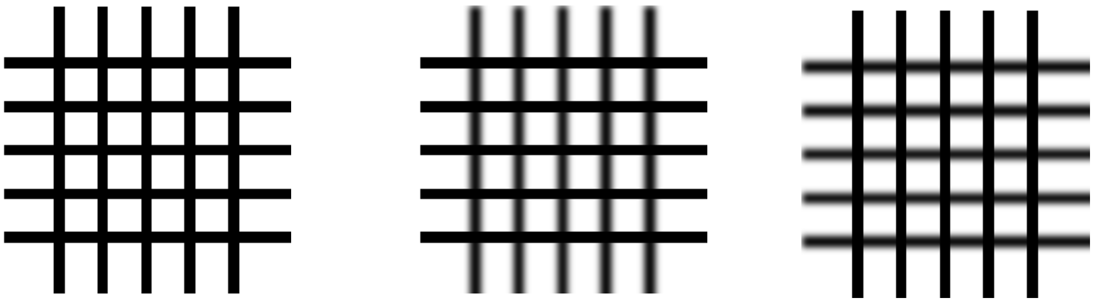
- Coma is the aberration resulting in varying focal length for rays which originate *off the axis* of the lens
  - Similar to spherical aberration, but for off-axis object/image points
- Rim rays come to focus nearer the axis than do rays passing near the center of the lens
  - a transverse aberration
- Coma results in a clearly defined geometrical shape (not a general blur) – comet-shaped (or teardrop-shaped)
- Proportional to the square of the aperture

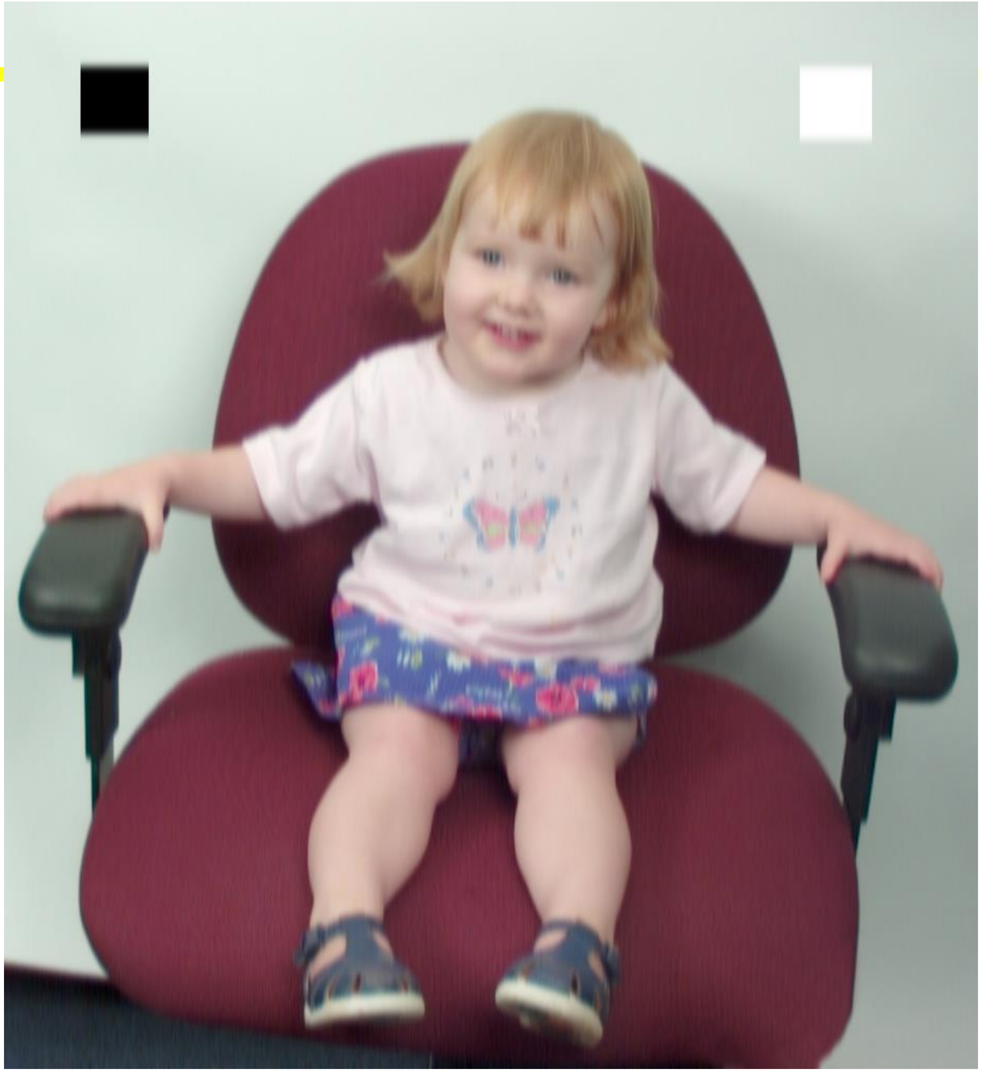




Astigmatism

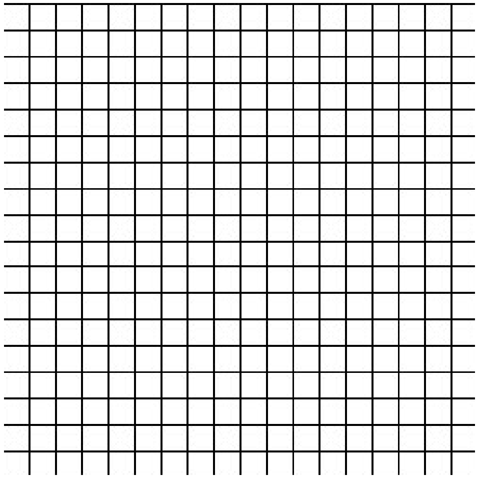
- Astigmatism is caused by the lens not having a perfectly spherical shape
  - Different curvature in different cross-sections (e.g., a football or a boat hull)
  - Horizontal and vertical edges/stripes focus differently
  - Result of poor lens grinding



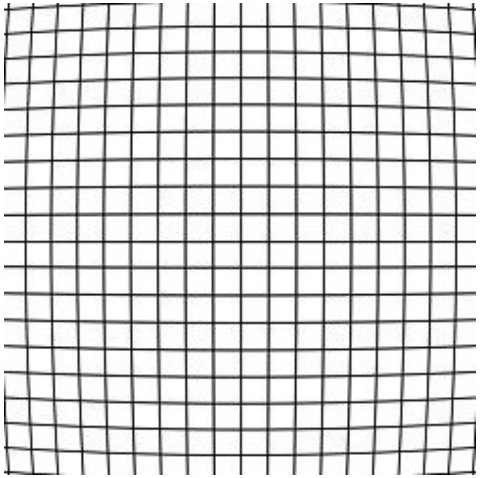


Radial distortion

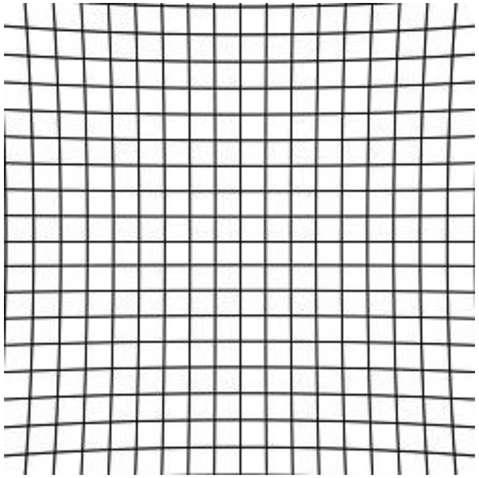
- Variation in the magnification for object points at different distances from the optical axis
- Effect increases with distance from the optical axis
- Straight lines become bent!
- Two main descriptions: *barrel* distortion and *pincushion* distortion
- Can be modeled and corrected for



Correct



Barrel



Pincushion

University of California
Santa Barbara

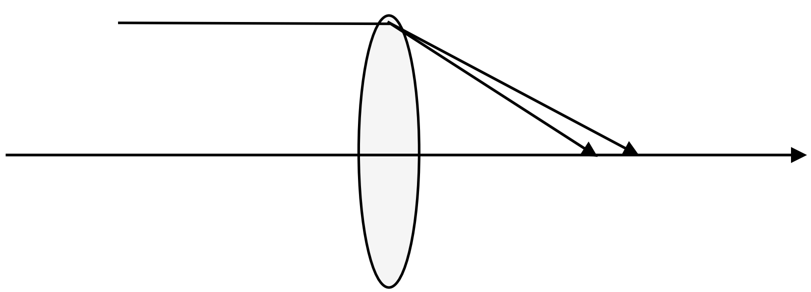
Correcting for radial distortion

Original

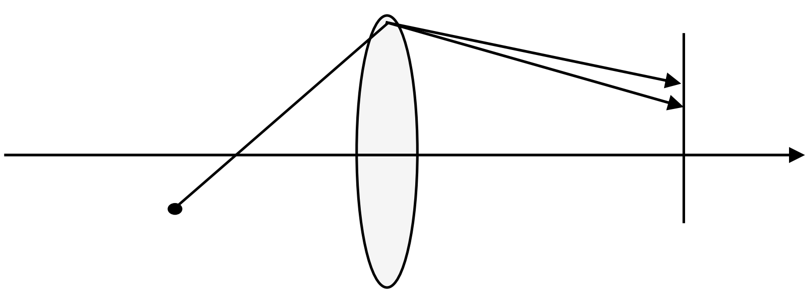
Corrected

Chromatic Aberration

- The index of refraction varies with wavelength (called *dispersion*)
  - Shorter wavelengths are bent more
- This causes **longitudinal** chromatic aberration – shorter wavelength rays focus closer to the lens
- Another effect is **lateral** chromatic aberration – shorter wavelength rays focus closer to the optical axis
- Can be reduced by using two lenses together, convex and concave



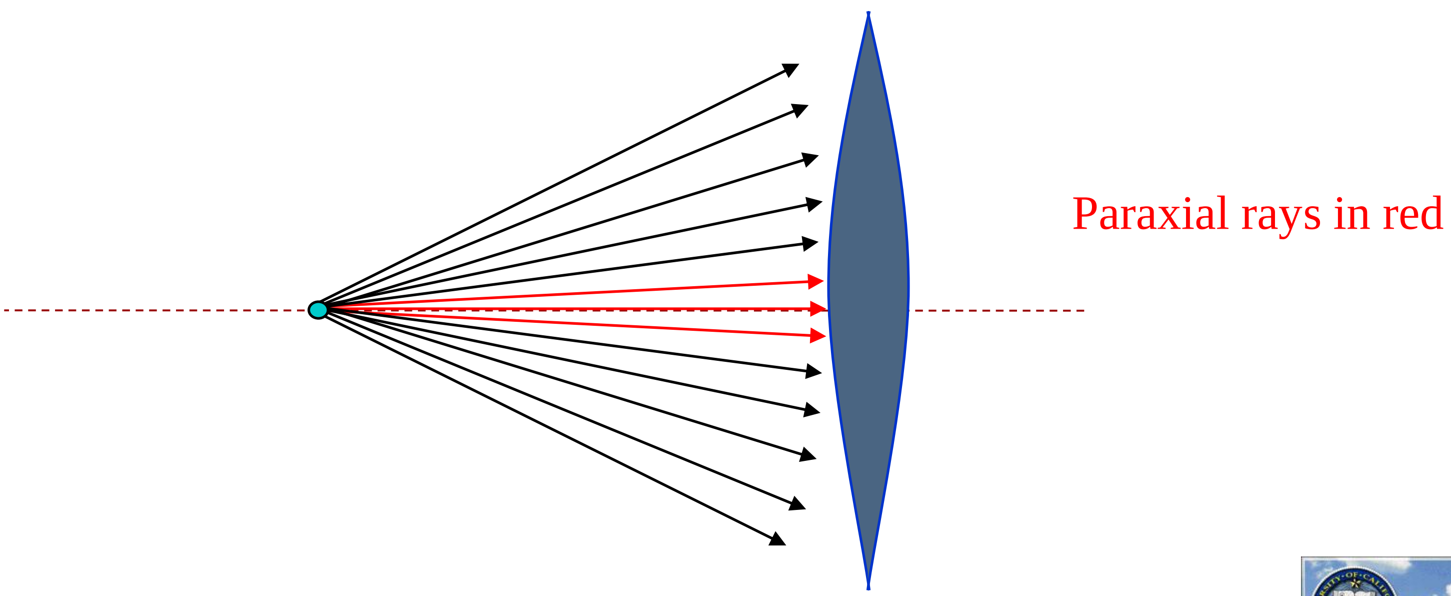
Longitudinal



Lateral

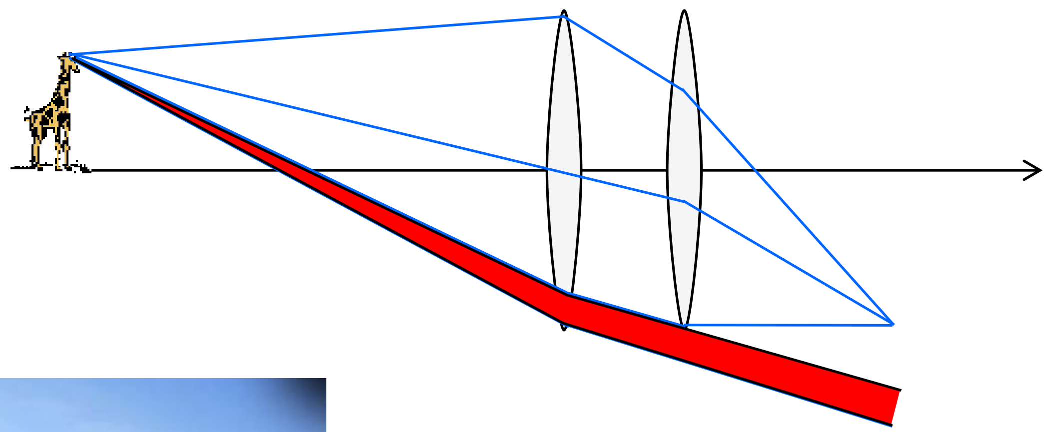
Paraxial rays

- Lenses are not perfect – they cause blurring
  - But in many cases, the blurring is minimal
  - Minimal blurring occurs when viewing only *paraxial rays*
- Paraxial rays – Rays which make a small angle with the central axis (in this case, perpendicular to the interface)



Vignetting

- Vignetting is the darkening of the corners of an image relative to its center, due to the use of multiple lenses



Lost rays

Vignetting

http://www.ptgui.com/examples/vigntutorial.html

http://www.tlucretius.net/Photo/eHolga.html

Image irradiance on the image plane

- The image irradiance (*E*) is proportional to the object radiance (*L*)

Angle off optical axis

Lens diameter

$$E = \left[\frac{\pi}{4} \left(\frac{d}{f} \right)^2 \cos^4 \alpha \right] L \quad = k(\alpha) L$$

Focus distance

What the image reports to us via pixel values

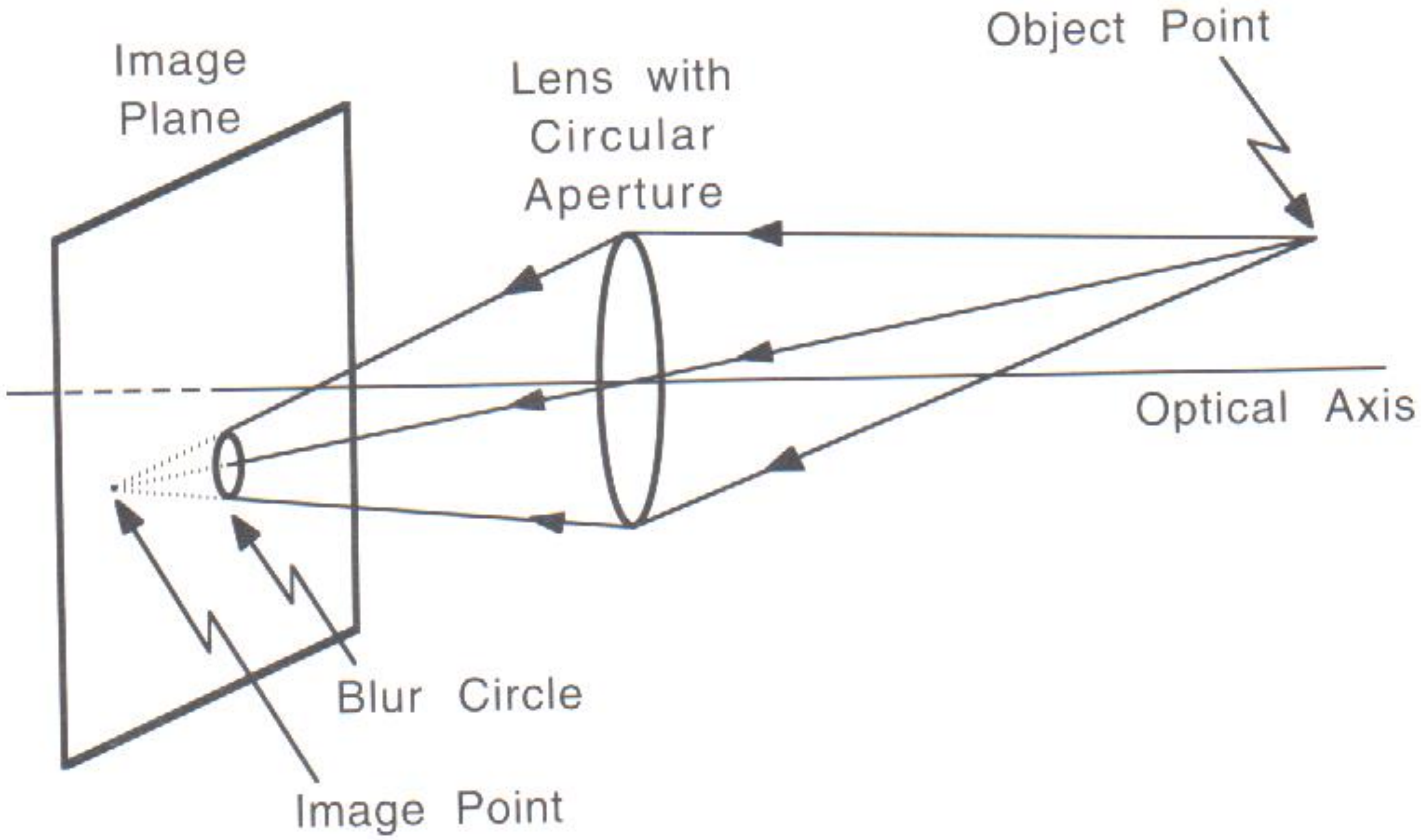
What we really want to know

$\cos^4 \alpha$ – 4^{th} power contributes to vignetting!

Lens lessons

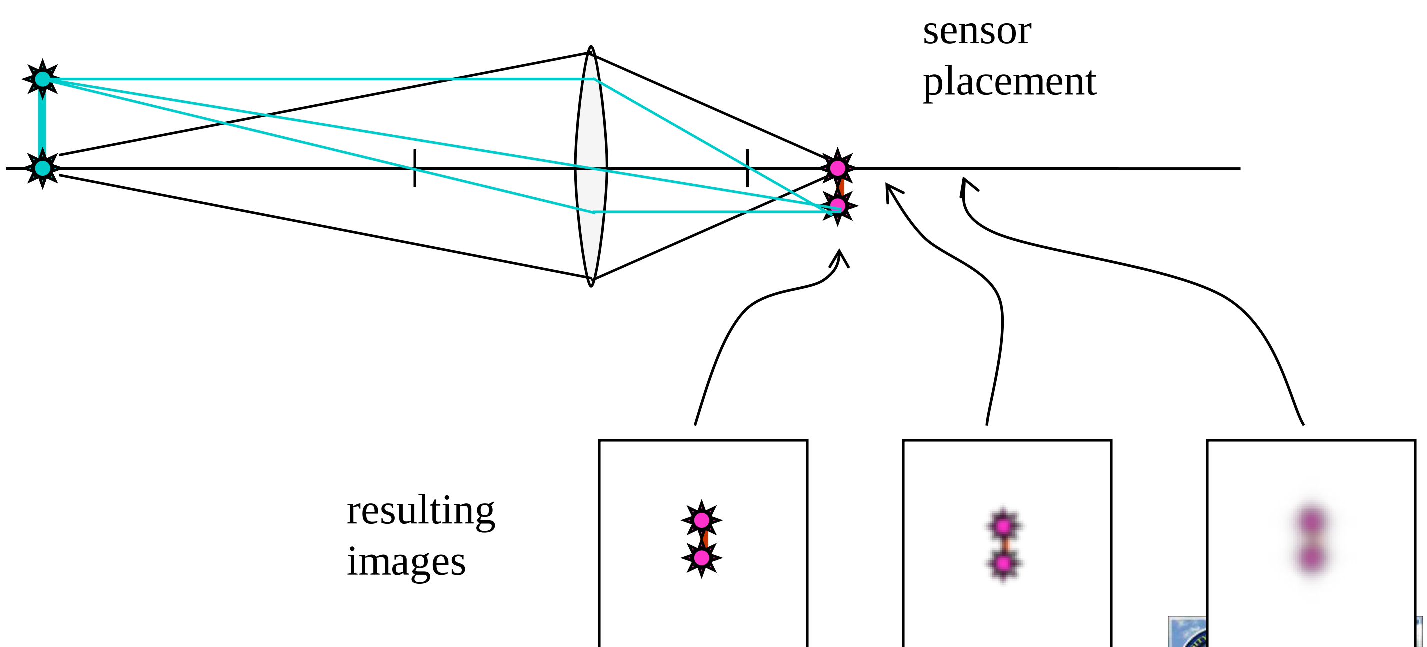
- No lens is perfect
- Even a perfectly-shaped lens is not perfect!
- Monochromatic light images best
- Good optical design (multiple lenses) and good craftsmanship (careful and precise lens grinding) can reduce aberration effects
- Paraxial rays are your best bet
- Some effects are only noticeable at high resolution

Blur due to misfocus



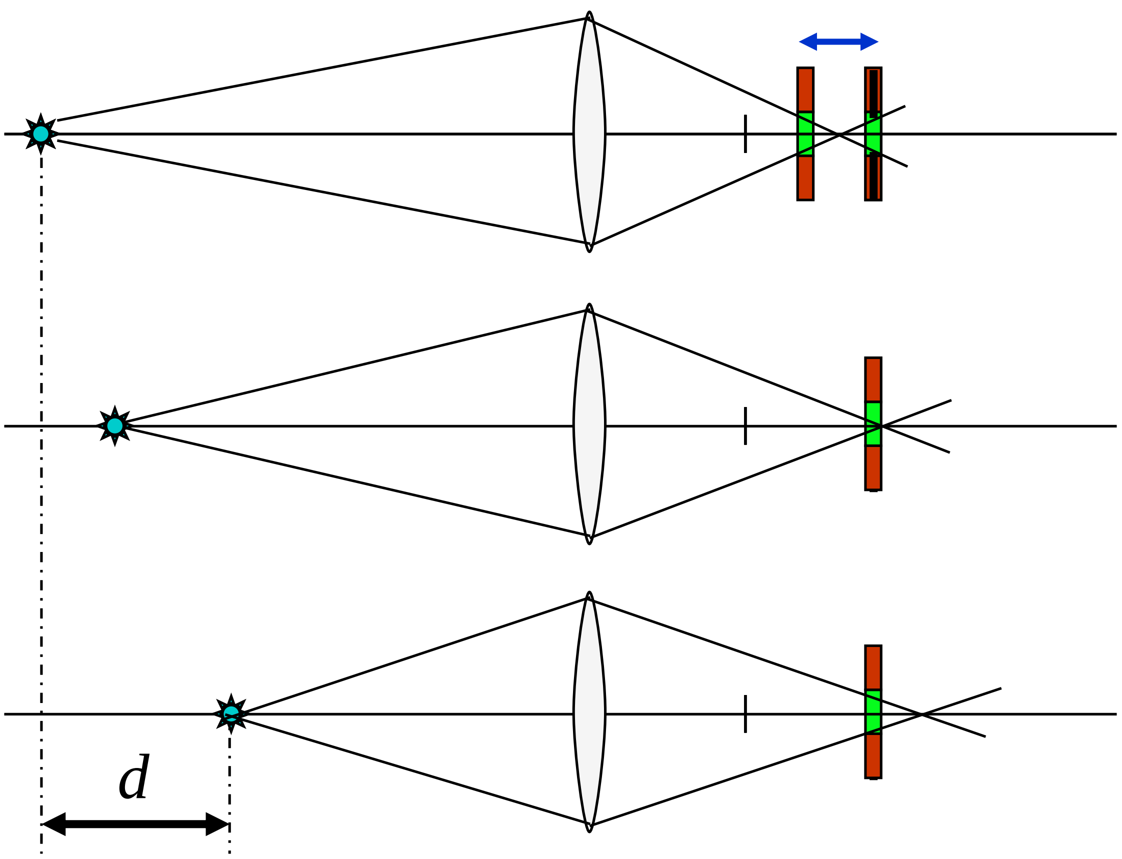
Depth of Focus/Field

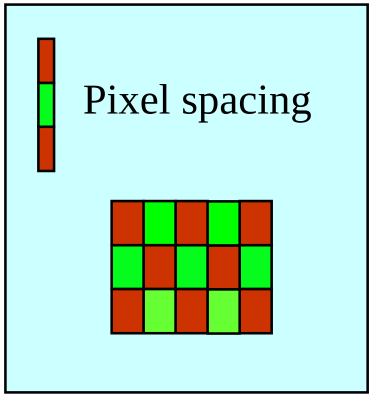
- In addition to lens aberrations, blur is caused by having the imager (sensor array) too close or too far – away from where the point is focused
- Depending on sensor resolution, small amounts of blur may not matter



Depth of Focus/Field

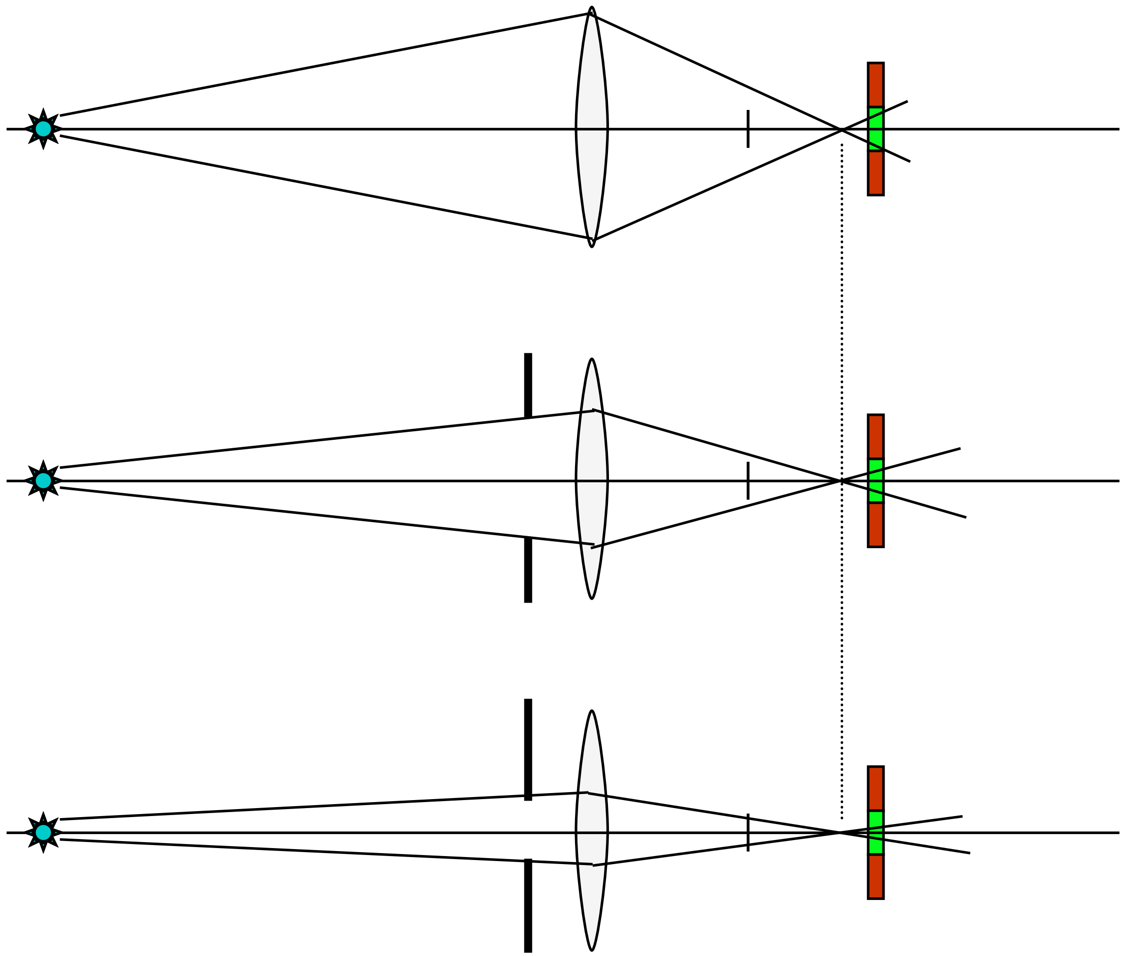
- Depth of **focus** – the distance of the imager along the axis where a single object point produces a focused image point
- Depth of **field** – the distance of the object point along the axis





Depth of Focus

- Aperture size affects depth of focus/field



Film Exposure

- Right amount of light depends on
  - Shutter speed – how long is the exposure
  - f/stop – how much light is allowed through the lens
  - Film speed – sensitivity of the film
- Shutter speed
  - Measured in seconds (or fraction of a second)
  - Neighboring setting either half or double the exposure time
  - 8,4,2,1,1/4,1/8,1/15,1/30,1/60,1/125,1/250,1/500,1/1000

F/stops

- Popular settings are f/1.4,f/2.0,f/2.8,f/4,f/5.6,f/8,f/11,f/16,f/22
- From brighter (f/1.4) to darker (f/22)
- Neighboring settings half or double the amount of light entering the camera
- focal length/iris diameter = f/stops
  - E.g. 50mm/25mm=f/2, so aperture diameter is half the focal length

F=50mm

f/stop	Diameter of aperture (mm)	Radius of aperture (mm)	Area of Aperture (sq. mm)
f/1.0	50.0	25.0	1,963
f/1.4	35.7	17.9	1,002
f/2.0	25.0	12.5	491
f/2.8	17.9	8.9	250
f/4	12.5	6.3	123
f/5.6	8.9	4.5	63
f/8	6.3	3.1	31
f/11	4.5	2.3	16
f/16	3.1	1.6	8
f/22	2.3	1.1	4

Varying Focal Length

24mm

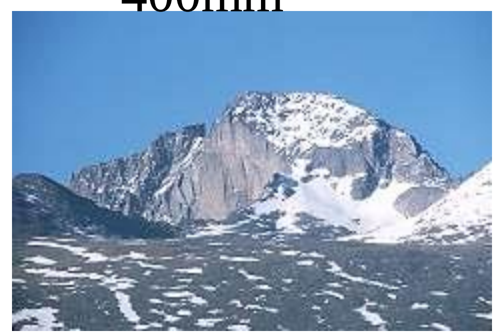
35mm

50mm

100mm

200mm

400mm



800mm

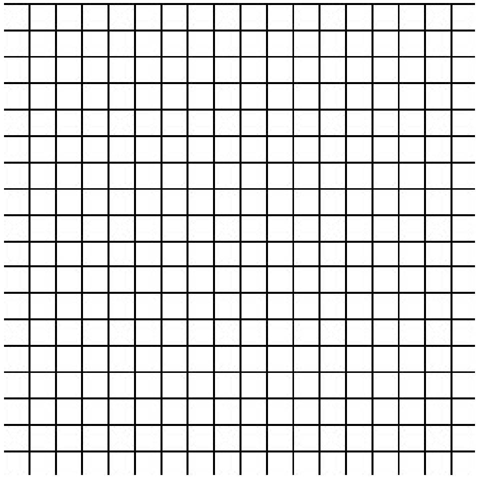
1200mm

Calibrating for radial distortion

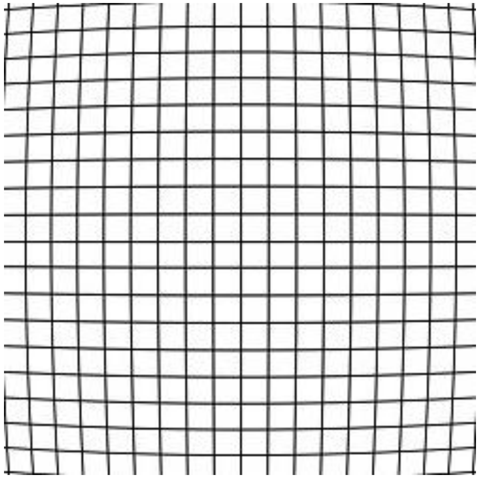
- The camera lens also introduces errors of several type (as we've already discussed):
  - Spherical aberration
  - Coma
  - Chromatic aberration
  - Vignetting
  - Astigmatism
  - Misfocus
  - Radial distortion
- Of these, *radial distortion* is the most significant in most systems, and it can be corrected for

Radial distortion

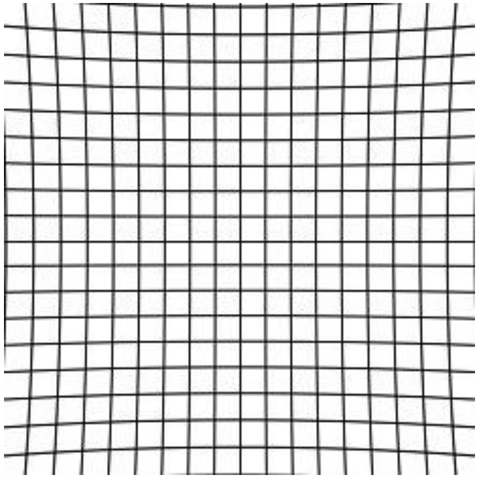
- Variation in the magnification for object points at different distances from the optical axis
- Effect increases with distance from the optical axis
- Straight lines become bent!
- Two main descriptions: *barrel* distortion and *pincushion* distortion
- Can be modeled and corrected for



Correct



Barrel



Pincushion

University of California Santa Barbara

Correcting for radial distortion

Original

Corrected

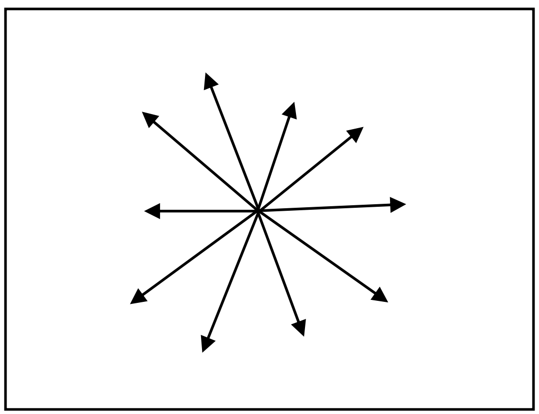
Modeling Lens Distortion

- Radial, Barrel, Pincushion, etc.
  - Modeled as bi-cubic (or bi-linear) with more parameters to compensate for
  - Very hard to solve

$$x_{real} = \begin{bmatrix} x_{ideal}^3 & x_{ideal}^2 & x_{ideal} & 1 \end{bmatrix} \begin{bmatrix} a_{11}^x & a_{12}^x & \cdots & a_{14}^x \\ a_{21}^x & a_{22}^x & \cdots & a_{24}^x \\ \cdots & \cdots & \cdots & \cdots \\ a_{41}^x & a_{42}^x & \cdots & a_{44}^x \end{bmatrix} \begin{bmatrix} y_{ideal}^3 \\ y_{ideal}^2 \\ y_{ideal} \\ 1 \end{bmatrix}$$

Modeling radial distortion

- The radial distortion can be modeled as a polynomial function (λ) of d^2, where d is the distance between the image center and the image point
  - Called the *radial alignment constraint*



$$\begin{bmatrix} u_d \\ v_d \end{bmatrix} = \lambda(d) \begin{bmatrix} u \\ v \end{bmatrix}$$

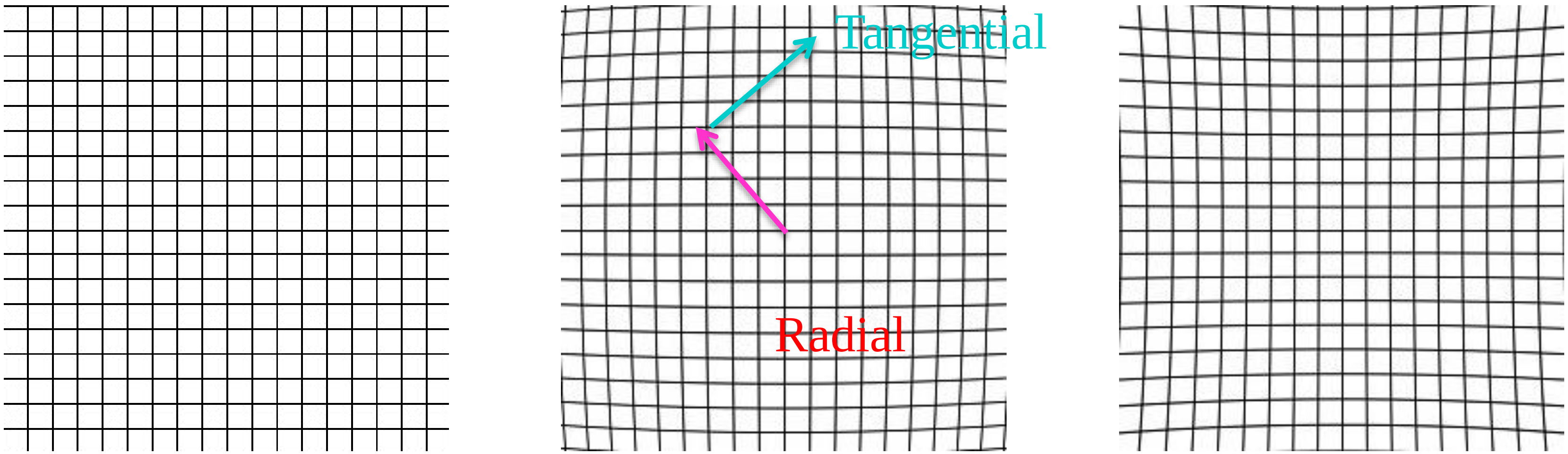
e.g., $\lambda(d) = 1 + \kappa_1 d^2 + \kappa_2 d^4 + \kappa_3 d^6$

q distortion coefficients ($q < 4$)

Less Frequently Used – Tangential Distortion

- Less common

$$dx = \begin{bmatrix} 2\,kc(3)\,x\,y + kc(4)\,(r^2 + 2x^2) \\ kc(3)\,(r^2 + 2y^2) + 2\,kc(4)\,x\,y \end{bmatrix}$$



Modeling distortion

- Since d is a function of u and v, we could also write λ as $\lambda(u, v)$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} \frac{1}{\lambda(u', v')} & 0 & 0 \\ 0 & \frac{1}{\lambda(u', v')} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{P}$$

where (u', v') comes from **K[R|T]P**

- Now do calibration by estimating the **11+*q*** parameters