6.869 Advances in Computer Vision

Spring 2011

Bill Freeman and Antonio Torralba

Lecture 11

MRF's (continued), cameras and lenses.

remember correction on Gibbs sampling

Motion application

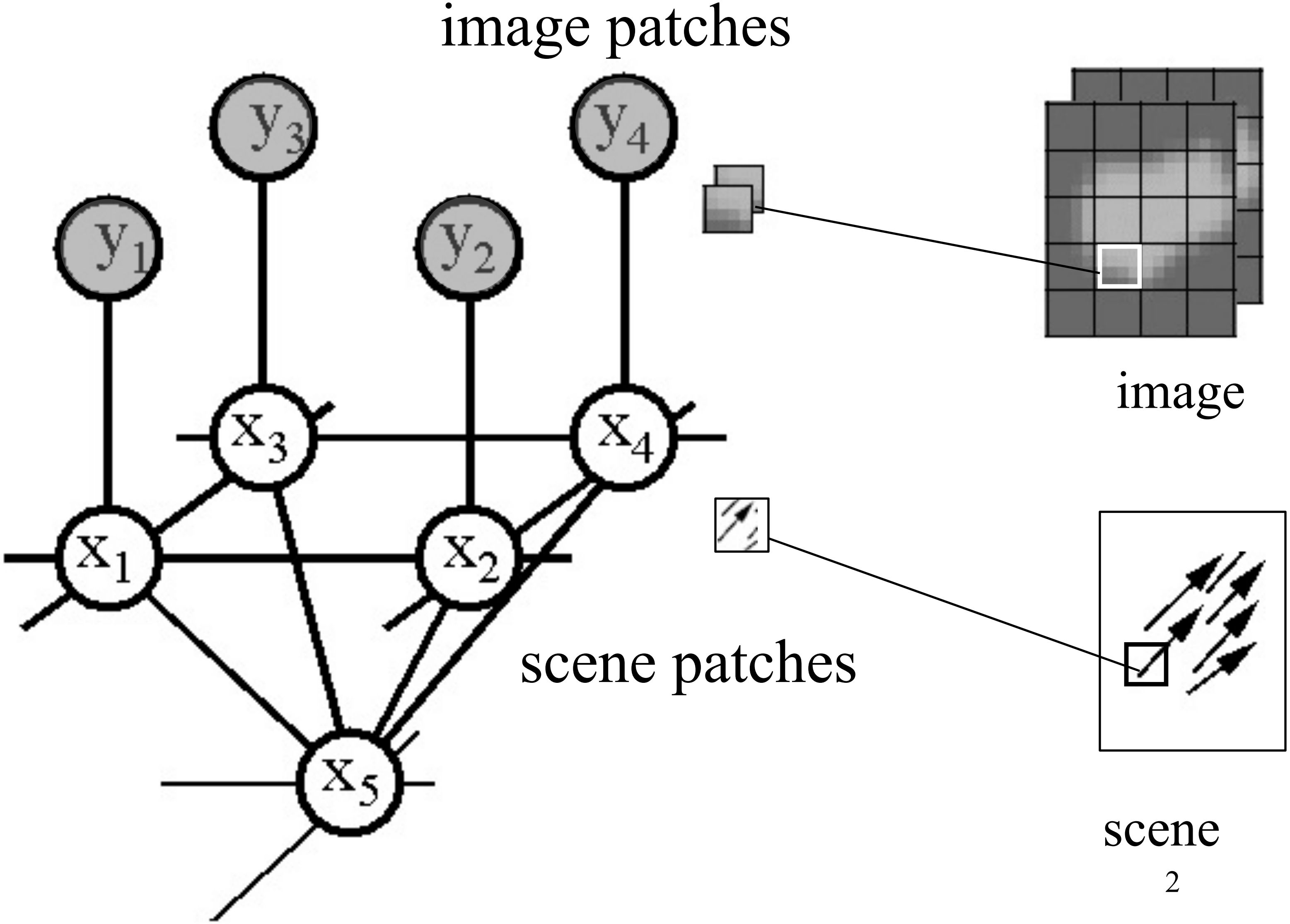


image patches

image

scene patches

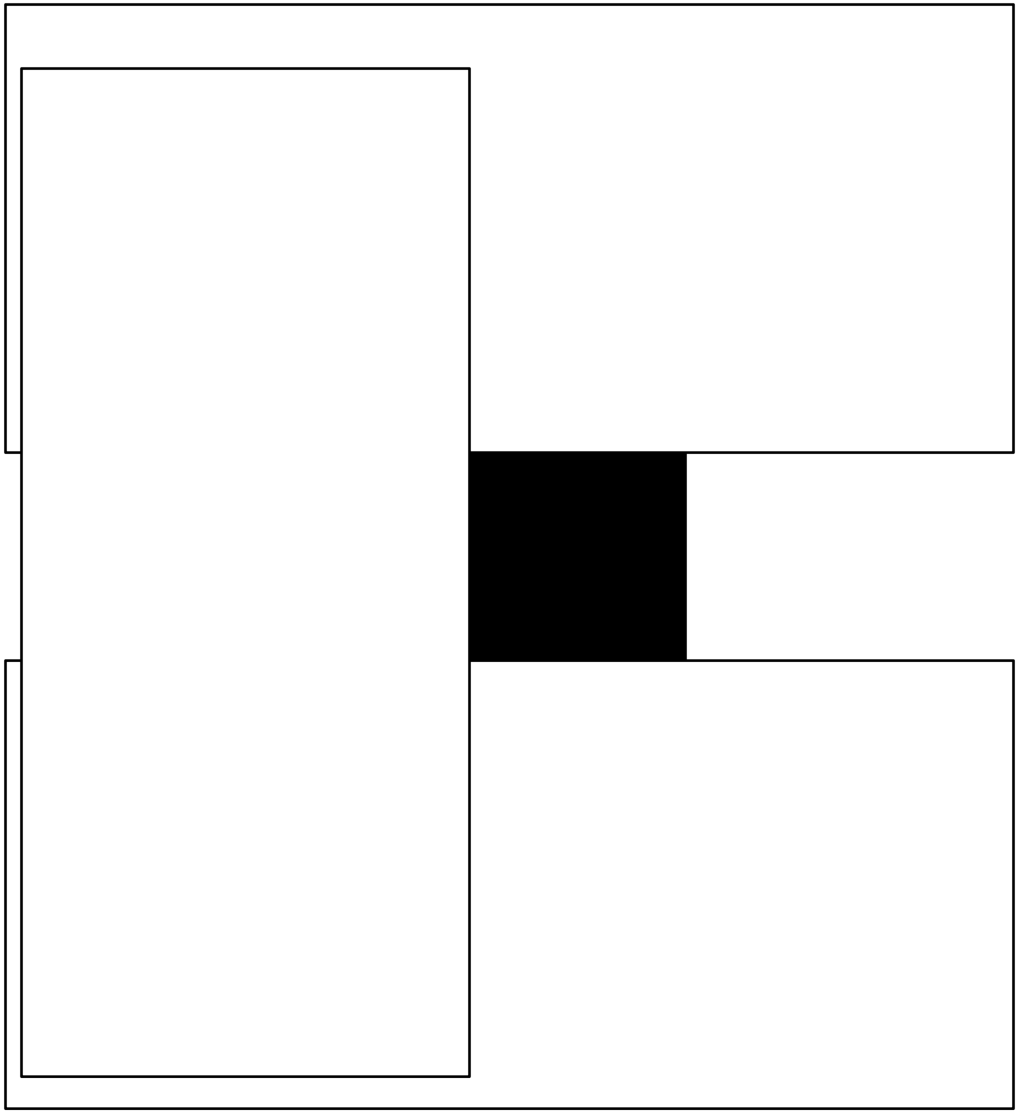
scene

What behavior should we see in a motion algorithm?

- Aperture problem
- Resolution through propagation of information
- Figure/ground discrimination

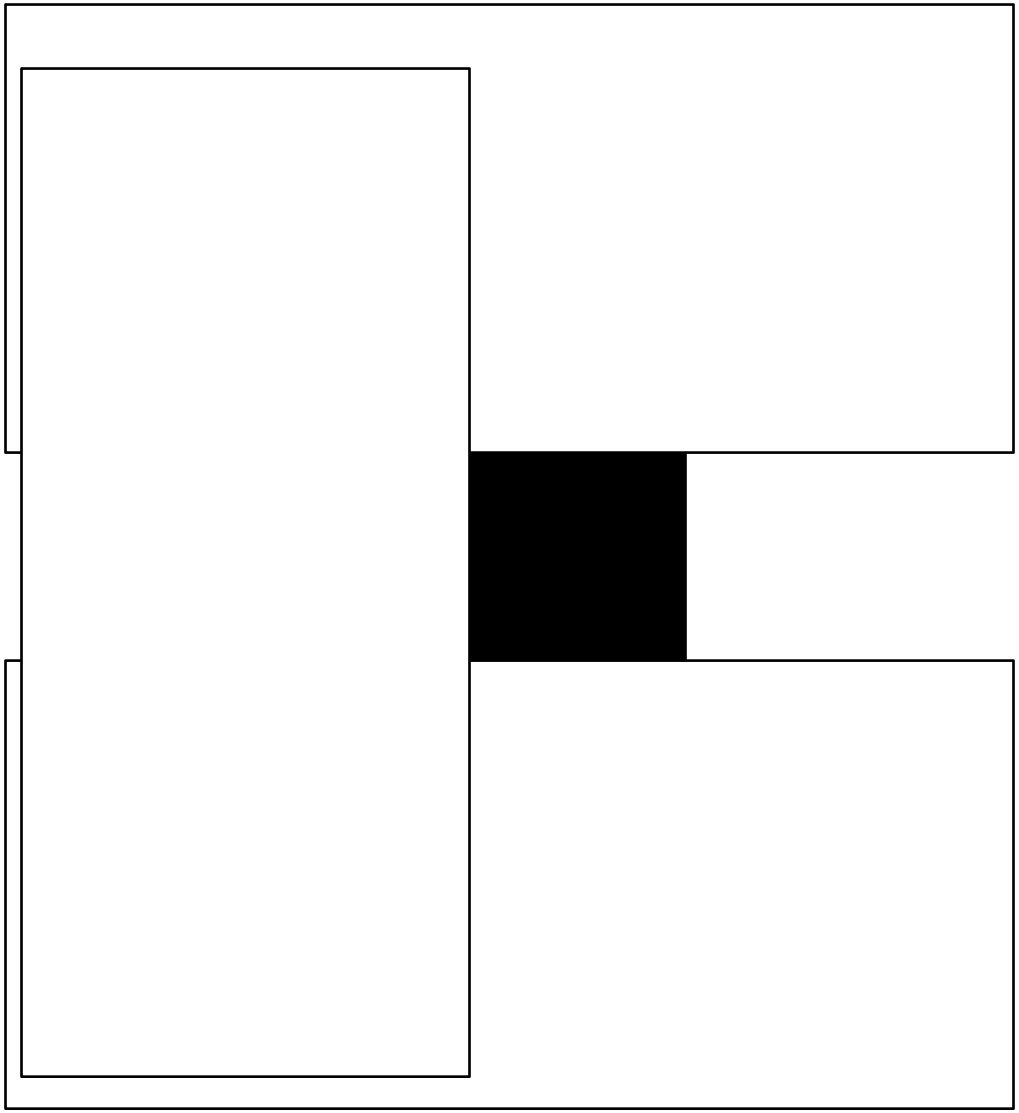
The aperture problem

http://web.mit.edu/persci/demos/Motion&Form/demos/one-square/one-square.html

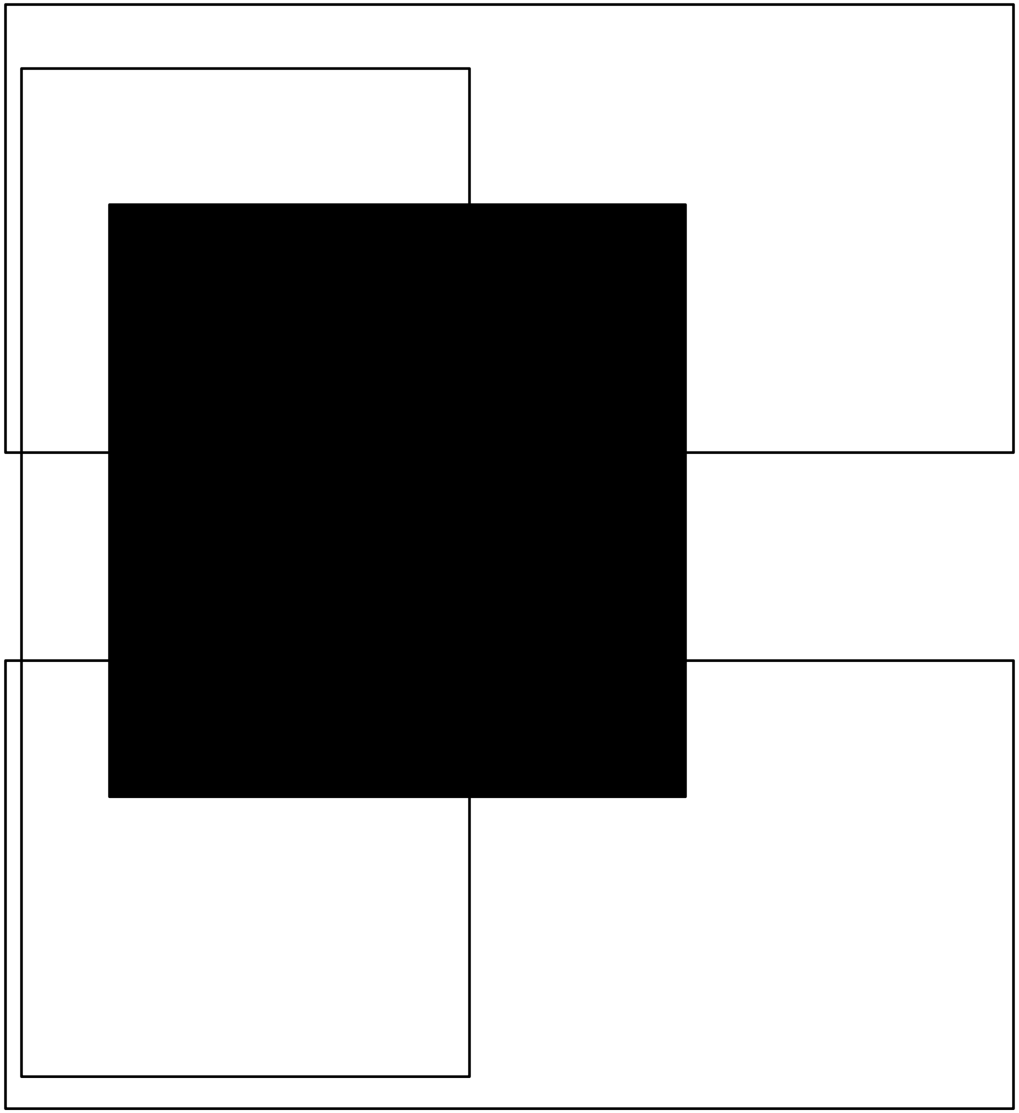


The aperture problem

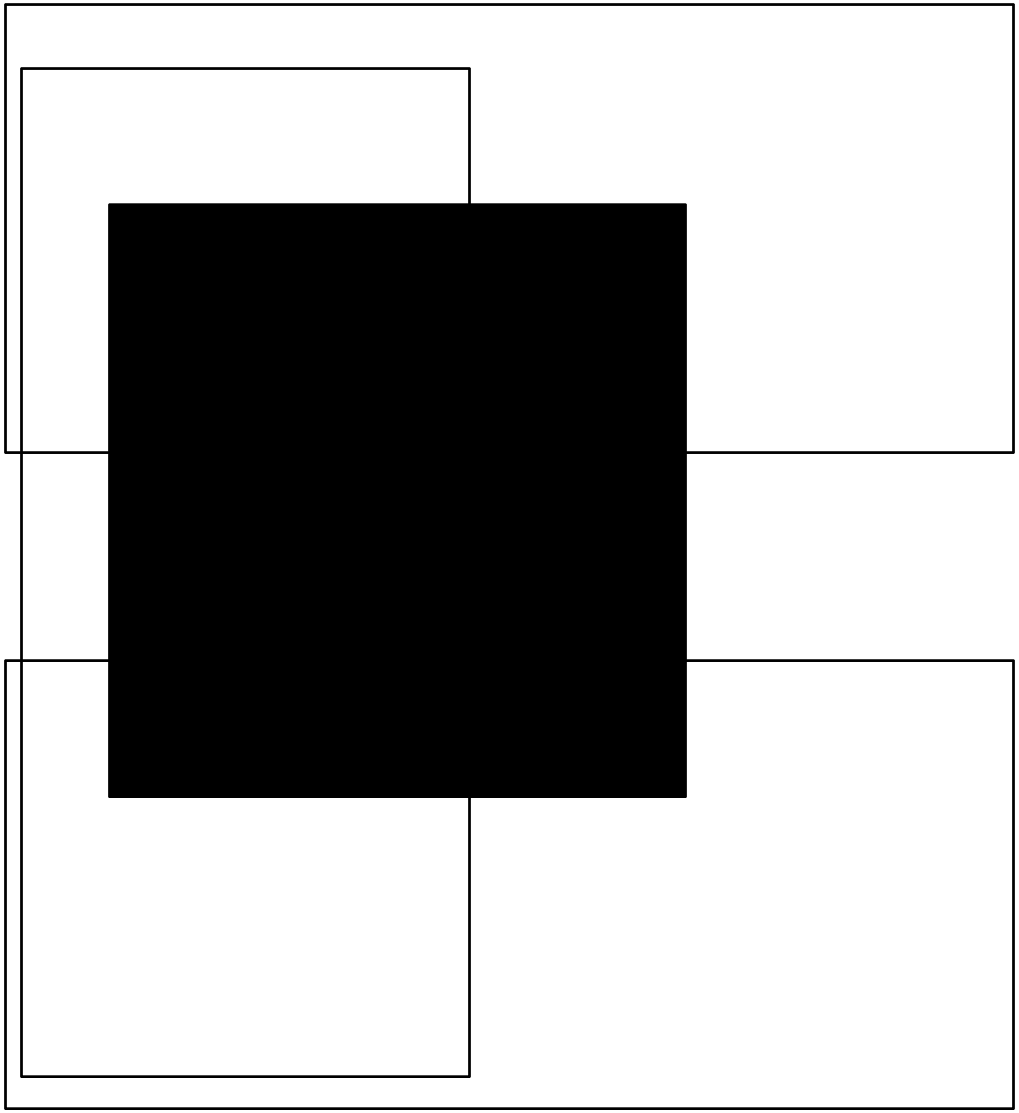
http://web.mit.edu/persci/demos/Motion&Form/demos/one-square/one-square.html



The aperture problem



The aperture problem



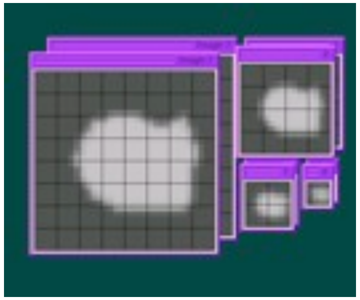
motion program demo

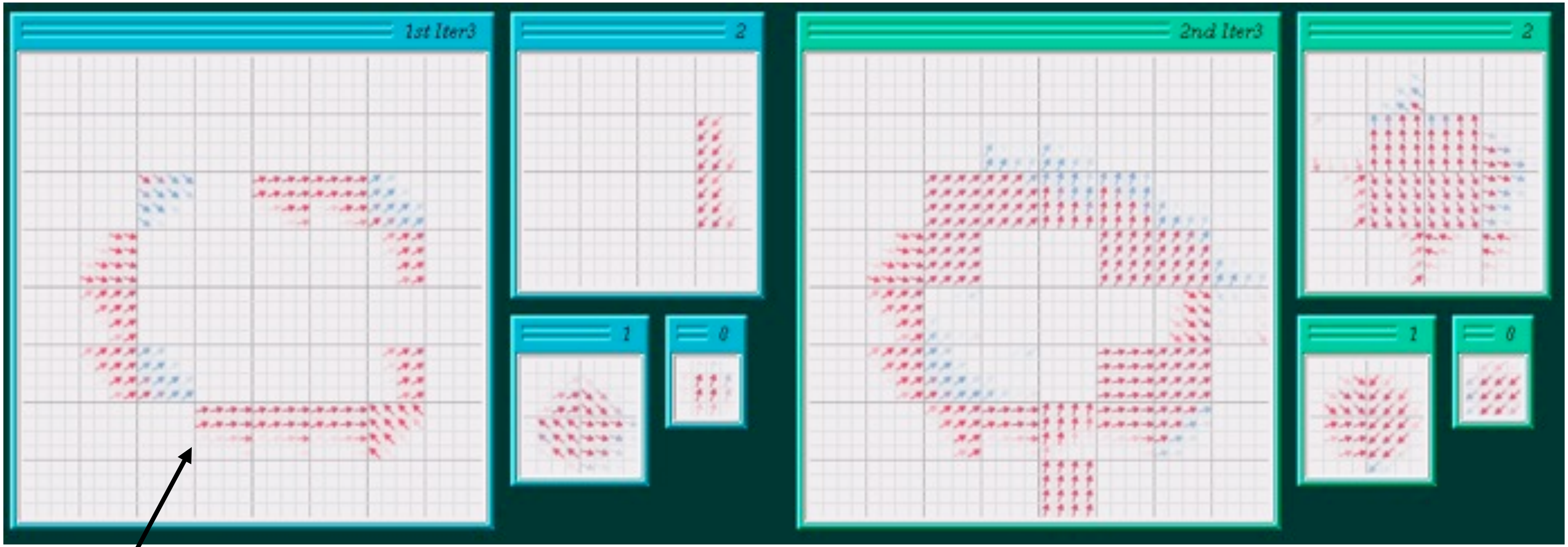
Inference:

Motion estimation results

(maxima of scene probability distributions displayed)

Image data



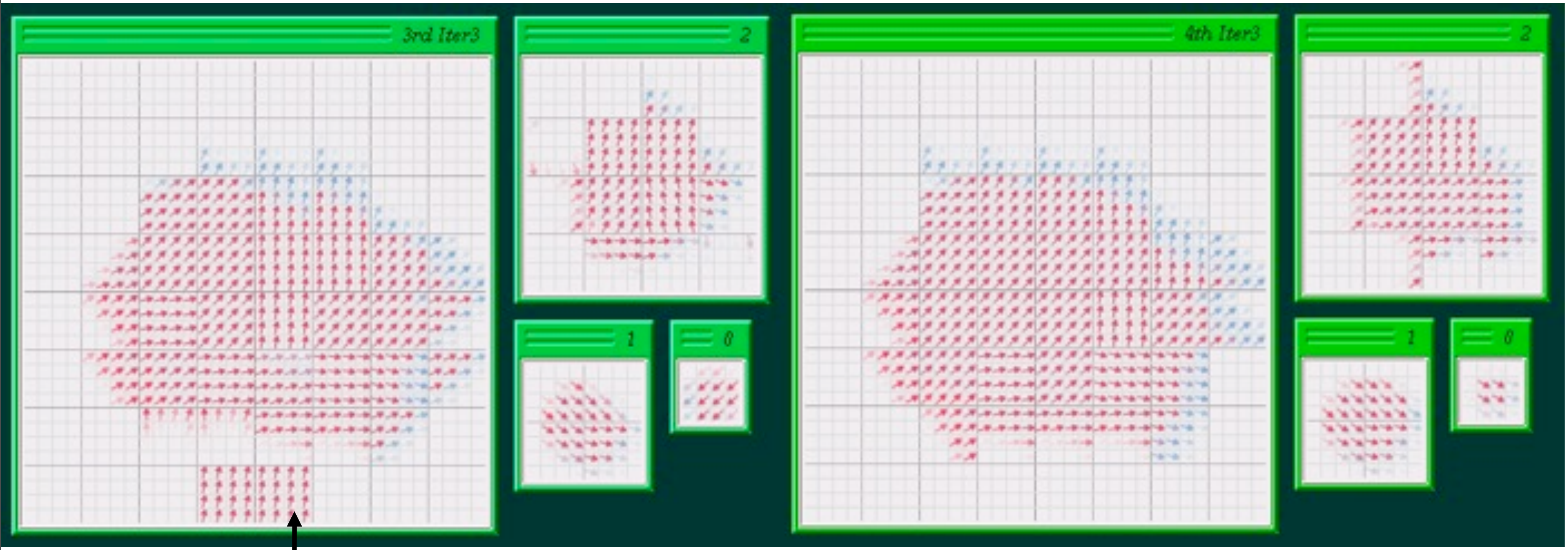


Iterations 0 and 1

Initial guesses only
show motion at edges.

Motion estimation results

(maxima of scene probability distributions displayed)

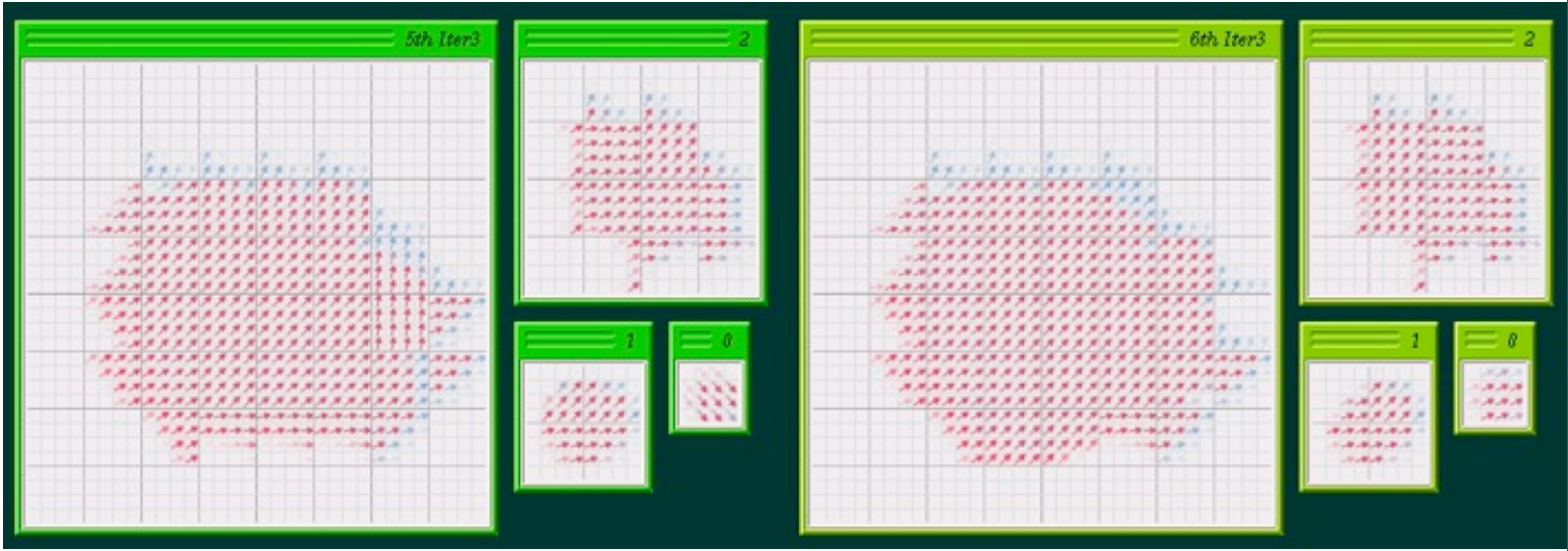


Iterations 2 and 3

Figure/ground still
unresolved here.

Motion estimation results

(maxima of scene probability distributions displayed)



Iterations 4 and 5

Final result compares well with vector quantized true (uniform) velocities.

Vision applications of MRF's

- Stereo
- Motion estimation
- Labelling shading and reflectance
- Segmentation
- Many others…

Random Fields for segmentation

I = Image pixels (observed)
h = foreground/background labels (hidden) – one label per pixel
θ = Parameters

$$\underbrace{p(h \mid I, \theta)}_{\textbf{Posterior}}$$

Random Fields for segmentation

I = Image pixels (observed)
h = foreground/background labels (hidden) – one label per pixel
θ = Parameters

$$\underbrace{p(h \mid I, \theta)}_{\textbf{Posterior}} \propto \underbrace{p(I, h \mid \theta)}_{\textbf{Joint}} = \underbrace{p(I \mid h, \theta)}_{\textbf{Likelihood}} \underbrace{p(h \mid \theta)}_{\textbf{Prior}}$$

Random Fields for segmentation

I = Image pixels (observed)
h = foreground/background labels (hidden) – one label per pixel
θ = Parameters

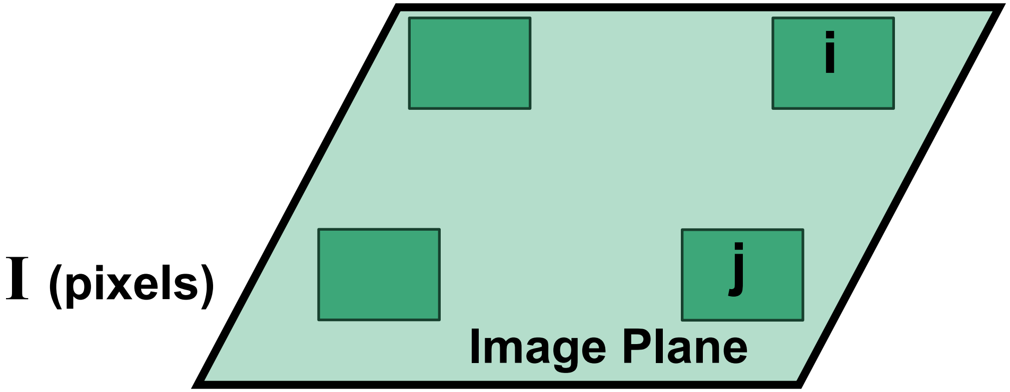
$$\underbrace{p(h \mid I,\theta)}_{\textbf{Posterior}} \propto \underbrace{p(I,h \mid \theta)}_{\textbf{Joint}} = \underbrace{p(I \mid h,\theta)}_{\textbf{Likelihood}} \underbrace{p(h \mid \theta)}_{\textbf{Prior}}$$

1. Generative approach models joint
  → Markov random field (MRF)

2. Discriminative approach models posterior directly
  → Conditional random field (CRF)

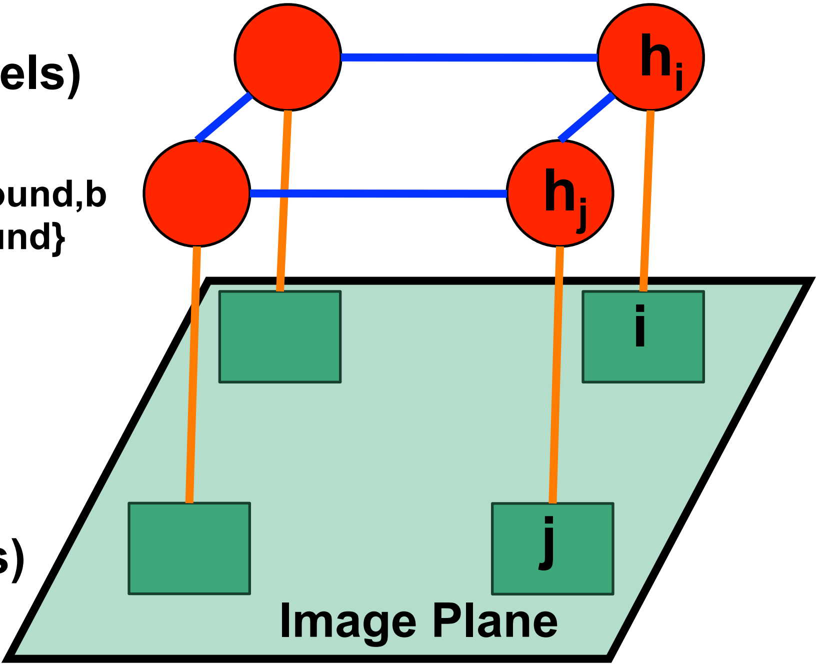
Generative Markov Random Field

$$p(h, I \mid \theta) = \boxed{p(I \mid h, \theta)}\boxed{p(h \mid \theta)}$$



Generative Markov Random Field

$$p(h, I \mid \theta) = \boxed{p(I \mid h, \theta)}\boxed{p(h \mid \theta)}$$



h **(labels)**

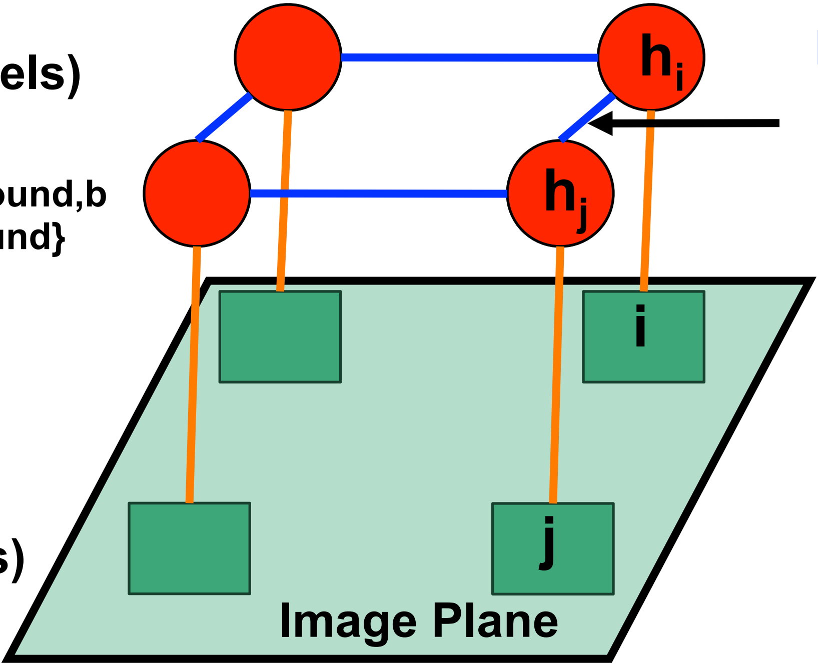
$\in$

{foreground,background}

I **(pixels)**

Generative Markov Random Field

$$p(h, I \mid \theta) = \boxed{p(I \mid h, \theta)}\,\boxed{p(h \mid \theta)}$$



h **(labels)**

$\in$ **{foreground,background}**

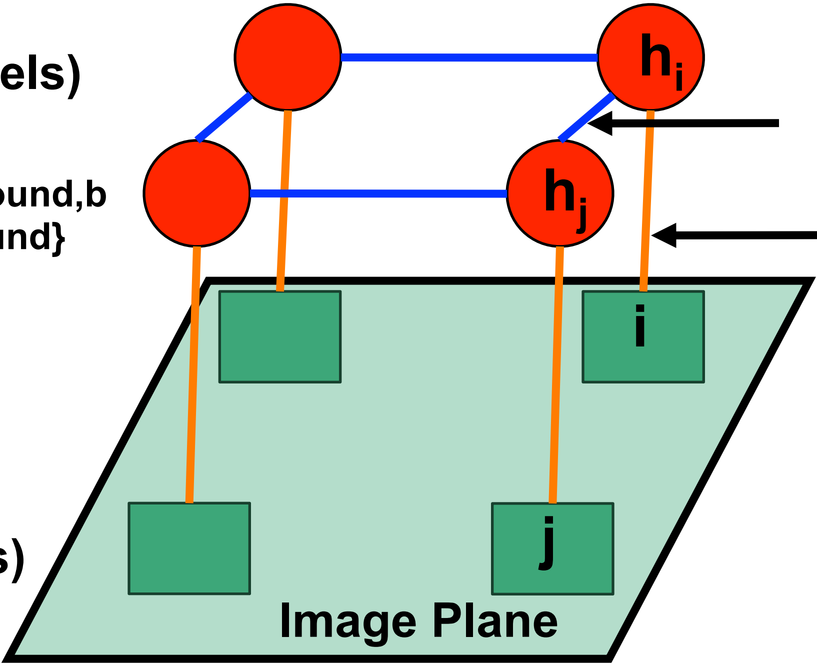
I **(pixels)**

Pairwise Potential (MRF)

$\psi_{ij}(h_i, h_j | \theta_{ij})$

Generative Markov Random Field

$$p(h, I \mid \theta) = \boxed{p(I \mid h, \theta)}\,\boxed{p(h \mid \theta)}$$



h (labels)

$\in$ {foreground,background}

I (pixels)

Pairwise Potential (MRF)

$\psi_{ij}(h_i, h_j|\theta_{ij})$

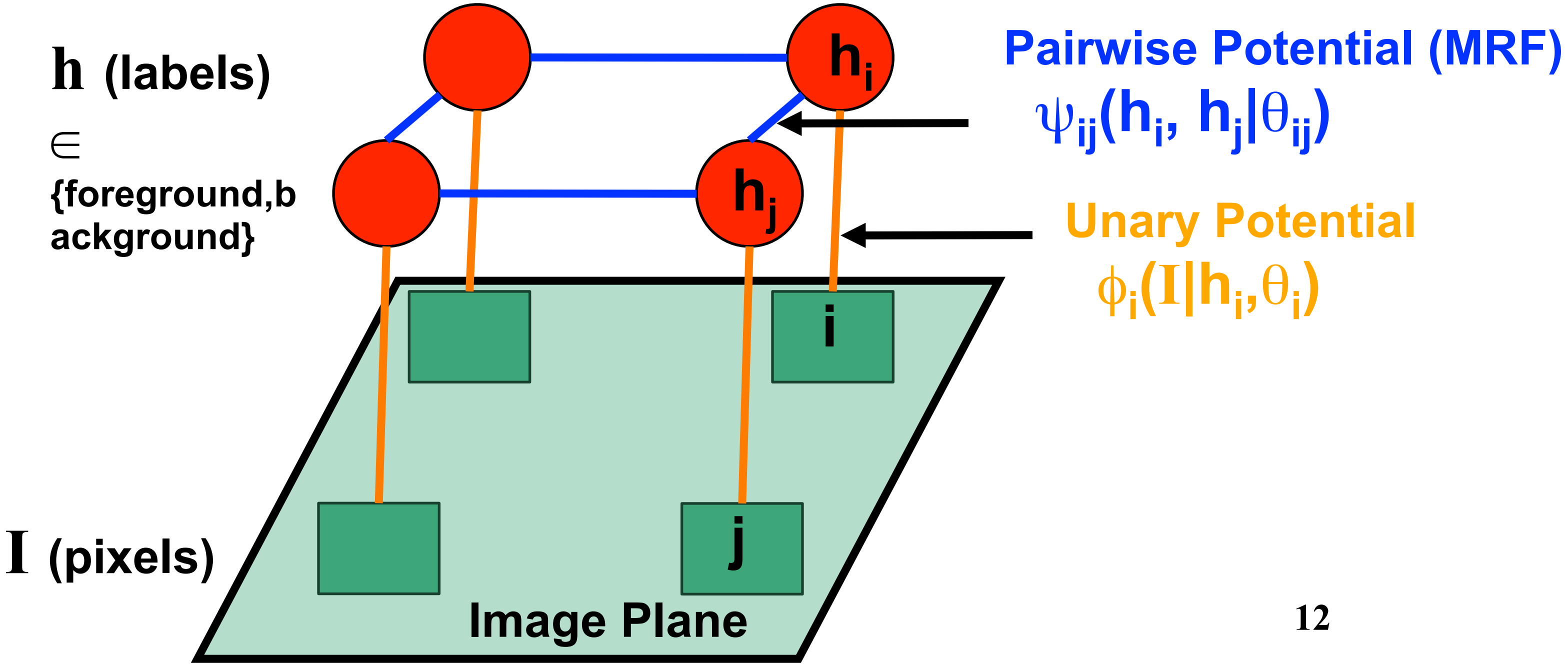
Unary Potential

$\phi_i(I|h_i,\theta_i)$

Generative Markov Random Field

$$p(h,I\,|\,\theta) = \boxed{p(I\,|\,h,\theta)}\boxed{p(h\,|\,\theta)}$$

$$= \frac{1}{Z(\theta)}\left[\prod_i \underbrace{\phi_i(I\,|\,h_i,\theta_i)}_{\text{Likelihood}} \prod_{ij} \underbrace{\psi_{ij}(h_i,h_j\,|\,\theta_{ij})}_{\text{MRF Prior}}\right]$$



h (labels) $\in$ {foreground,background}

I (pixels)

Image Plane

Pairwise Potential (MRF) $\psi_{ij}(h_i, h_j|\theta_{ij})$

Unary Potential $\phi_i(I|h_i,\theta_i)$

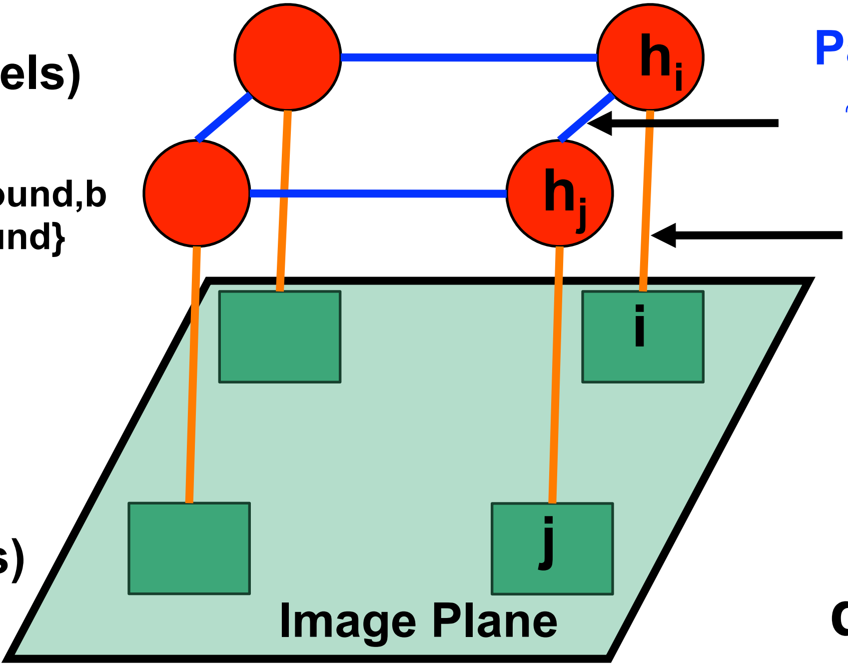
Generative Markov Random Field

$$p(h,I|\theta) = \boxed{p(I|h,\theta)}\boxed{p(h|\theta)}$$

$$= \frac{1}{Z(\theta)}\left[\prod_i \underbrace{\phi_i(I|h_i,\theta_i)}_{\text{Likelihood}}\prod_{ij}\underbrace{\psi_{ij}(h_i,h_j|\theta_{ij})}_{\text{MRF Prior}}\right]$$

h (labels)

$\in$ {foreground,background}



Pairwise Potential (MRF)

$\psi_{ij}(h_i, h_j|\theta_{ij})$

Unary Potential

$\phi_i(I|h_i,\theta_i)$

I (pixels)

Prior has no dependency on I

Conditional Random Field

Discriminative approach

Lafferty, McCallum and Pereira 2001

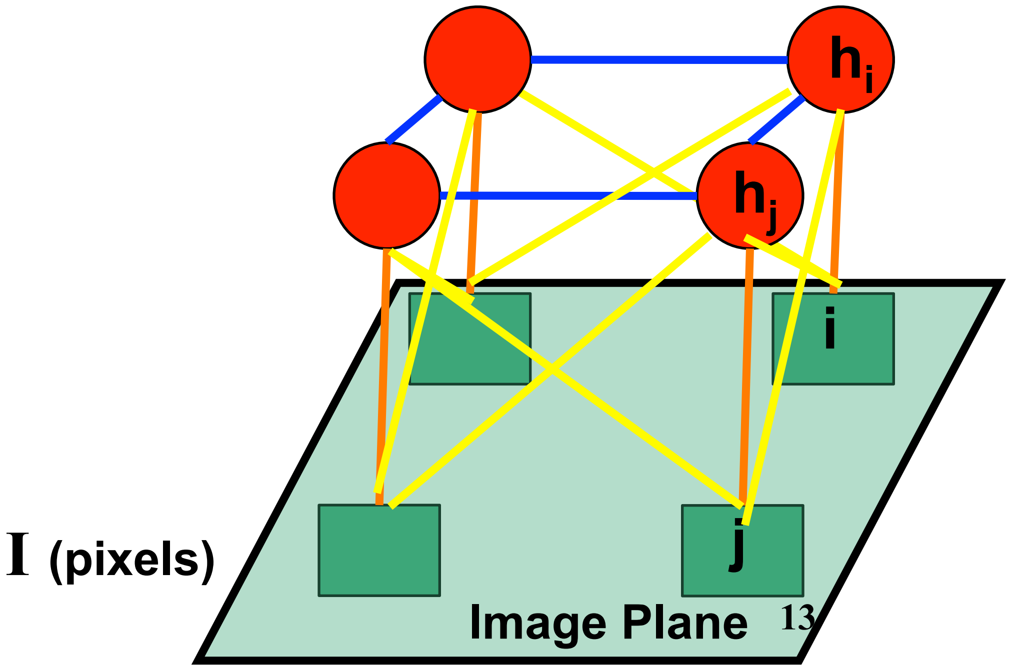
$$p(h \mid I, \theta) = \frac{1}{Z(I, \theta)} \left[\prod_i \underbrace{\phi_i(h_i, I \mid \theta_i)}_{\textbf{Unary}} \prod_{ij} \underbrace{\psi_{ij}(h_i, h_j \boxed{, I \mid} \theta_{ij})}_{\textbf{Pairwise}} \right]$$

Conditional Random Field

Discriminative approach

Lafferty, McCallum and Pereira 2001

$$p(h \mid I,\theta) = \frac{1}{Z(I,\theta)}\left[\prod_i \underbrace{\phi_i(h_i, I \mid \theta_i)}_{\text{Unary}} \prod_{ij} \underbrace{\psi_{ij}(h_i, h_j, I \mid \theta_{ij})}_{\text{Pairwise}}\right]$$



Conditional Random Field

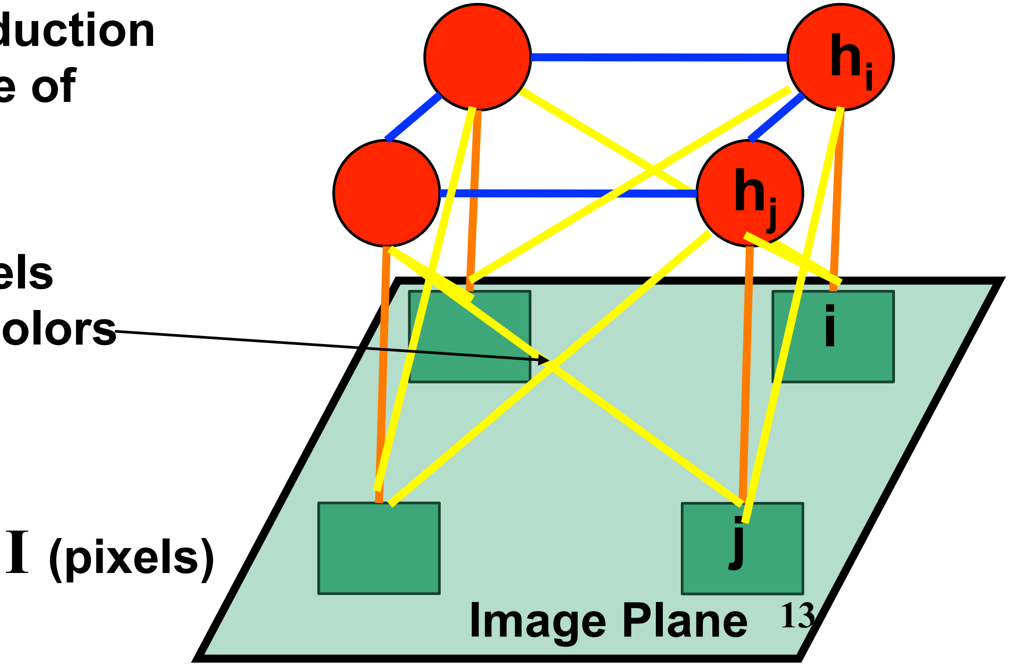
Discriminative approach

Lafferty, McCallum and Pereira 2001

$$p(h|I,\theta) = \frac{1}{Z(I,\theta)}\left[\prod_i \underbrace{\phi_i(h_i, I|\theta_i)}_{\text{Unary}} \prod_{ij} \underbrace{\psi_{ij}(h_i,h_j,I|\theta_{ij})}_{\text{Pairwise}}\right]$$

- **Dependency on I allows introduction of pairwise terms that make use of image.**

- **For example, neighboring labels should be similar only if pixel colors are**
similar → Contrast term



Conditional Random Field

Discriminative approach

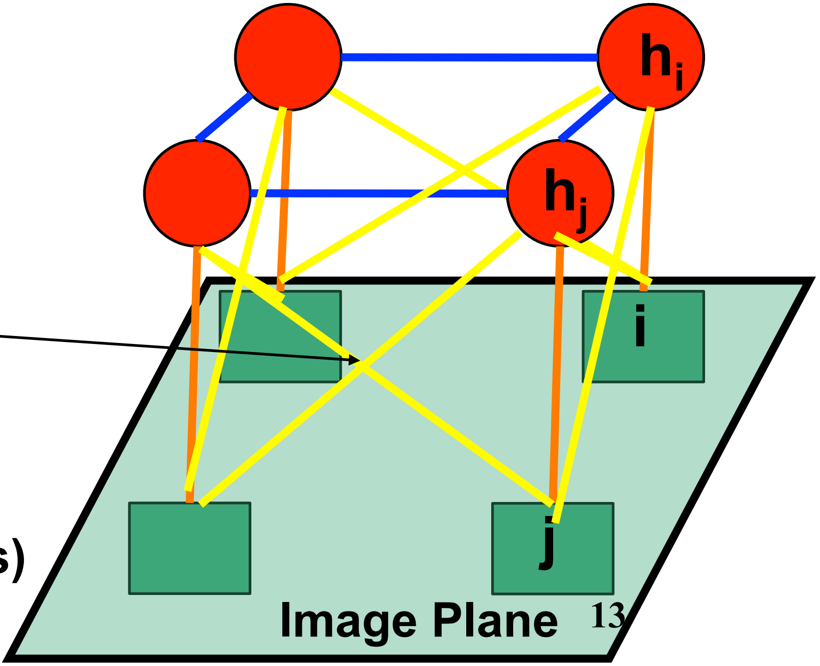
Lafferty, McCallum and Pereira 2001

$$p(h \mid I, \theta) = \frac{1}{Z(I,\theta)} \left[\prod_i \underbrace{\phi_i(h_i, I \mid \theta_i)}_{\textbf{Unary}} \prod_{ij} \underbrace{\psi_{ij}(h_i, h_j, I \mid \theta_{ij})}_{\textbf{Pairwise}} \right]$$

- **Dependency on I allows introduction of pairwise terms that make use of image.**

- **For example, neighboring labels should be similar only if pixel colors are similar → Contrast term**

e.g Kumar and Hebert 2003



OBJCUT

Kumar, Torr & Zisserman 2005

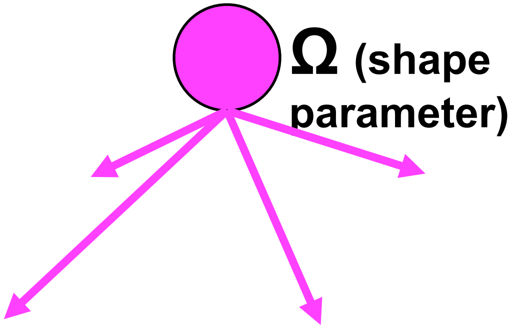
$$p(h\,|\,\Omega,I,\theta) \propto \left[\prod_i \overbrace{\underbrace{\phi_i^1(I\,|\,h_i,\theta_i)}_{\text{Color Likelihood}} \times \underbrace{\phi_i^2(h_i\,|\,\Omega)}_{\text{Distance from }\Omega}}^{\text{Unary}} \prod_{ij} \overbrace{\underbrace{\psi_{ij}^1(h_i,h_j\,|\,\theta_{ij})}_{\text{Label smoothness}} \times \underbrace{\psi_{ij}^2(I\,|\,h_i,h_j,\theta_{ij})}_{\text{Contrast}}}^{\text{Pairwise}} \right]$$

- **Ω is a shape prior on the labels from a Layered Pictorial Structure (LPS) model**

- **Segmentation by:**

  - Match LPS model to image (get number of samples, each with a different pose

  -Marginalize over the samples using a single graph cut [Boykov & Jolly, 2001]



OBJCUT

Kumar, Torr & Zisserman 2005

$$p(h \mid \Omega, I, \theta) \propto \left[\prod_i \overbrace{\underbrace{\phi_i^1(I \mid h_i, \theta_i)}_{\text{Color Likelihood}} \times \underbrace{\phi_i^2(h_i \mid \Omega)}_{\text{Distance from } \Omega}}^{\text{Unary}} \prod_{ij} \overbrace{\underbrace{\psi_{ij}^1(h_i, h_j \mid \theta_{ij})}_{\text{Label smoothness}} \times \underbrace{\psi_{ij}^2(I \mid h_i, h_j, \theta_{ij})}_{\text{Contrast}}}^{\text{Pairwise}} \right]$$

- **Ω is a shape prior on the labels from a Layered Pictorial Structure (LPS) model**

- **Segmentation by:**

  - Match LPS model to image (get number of samples, each with a different pose

  -Marginalize over the samples using a single graph cut [Boykov & Jolly, 2001]

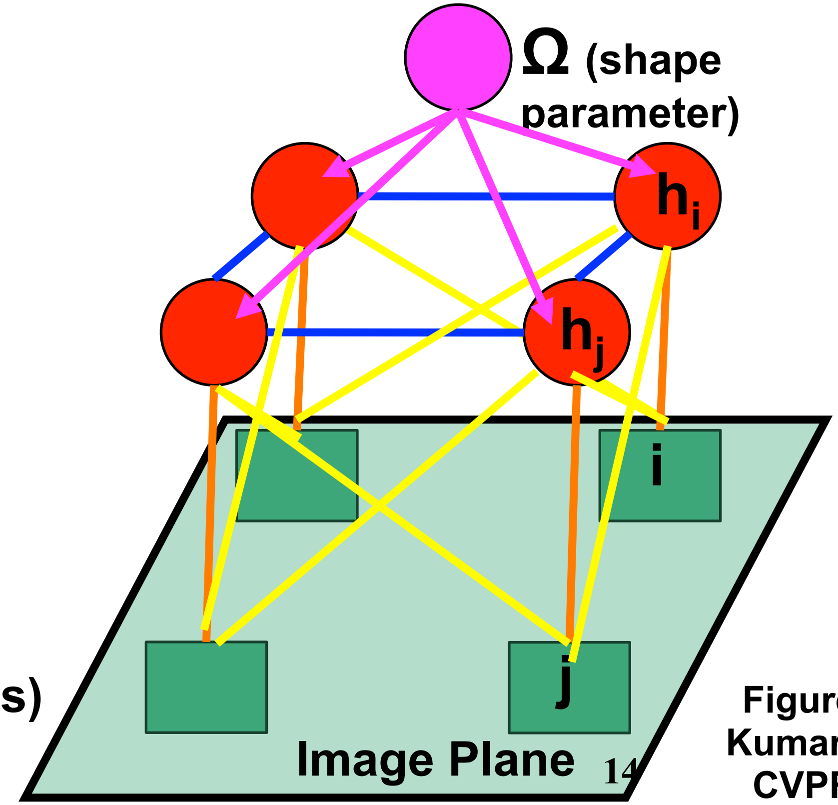
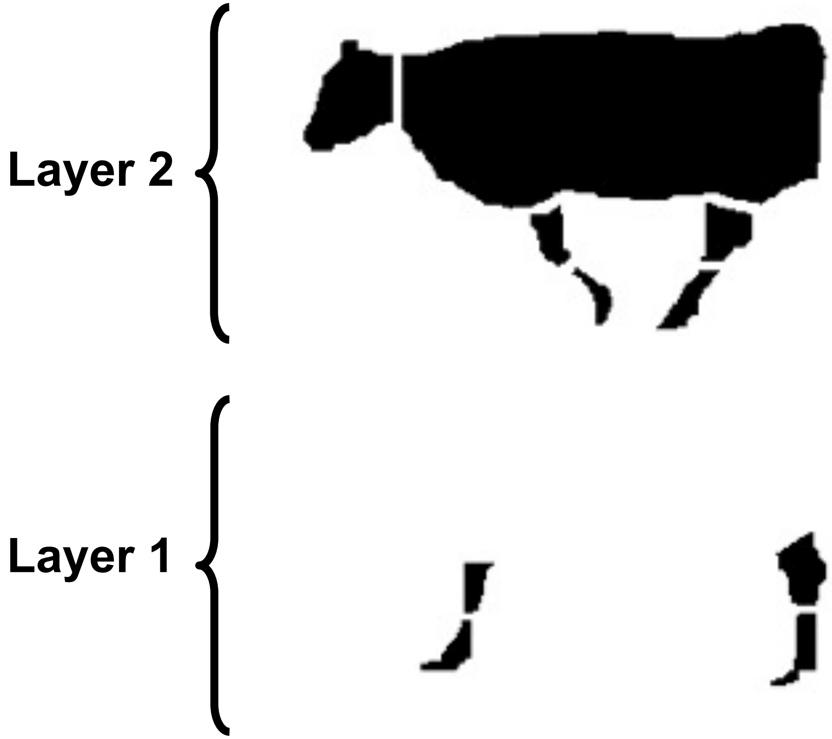
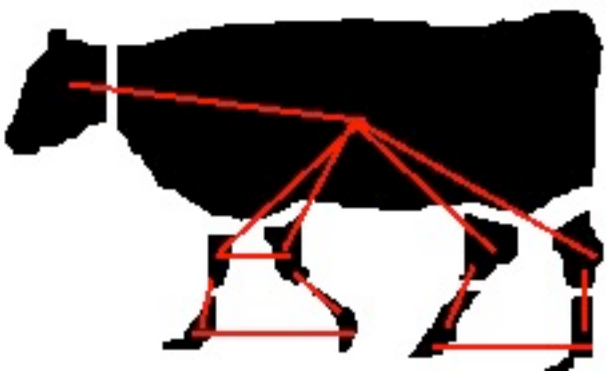


Figure from Kumar et al., CVPR 2005

OBJCUT:
Shape prior - **Ω** - Layered Pictorial Structures (LPS)

- Generative model
- Composition of parts + spatial layout





Spatial Layout
(Pairwise Configuration)

Parts in Layer 2 can occlude parts in Layer 1

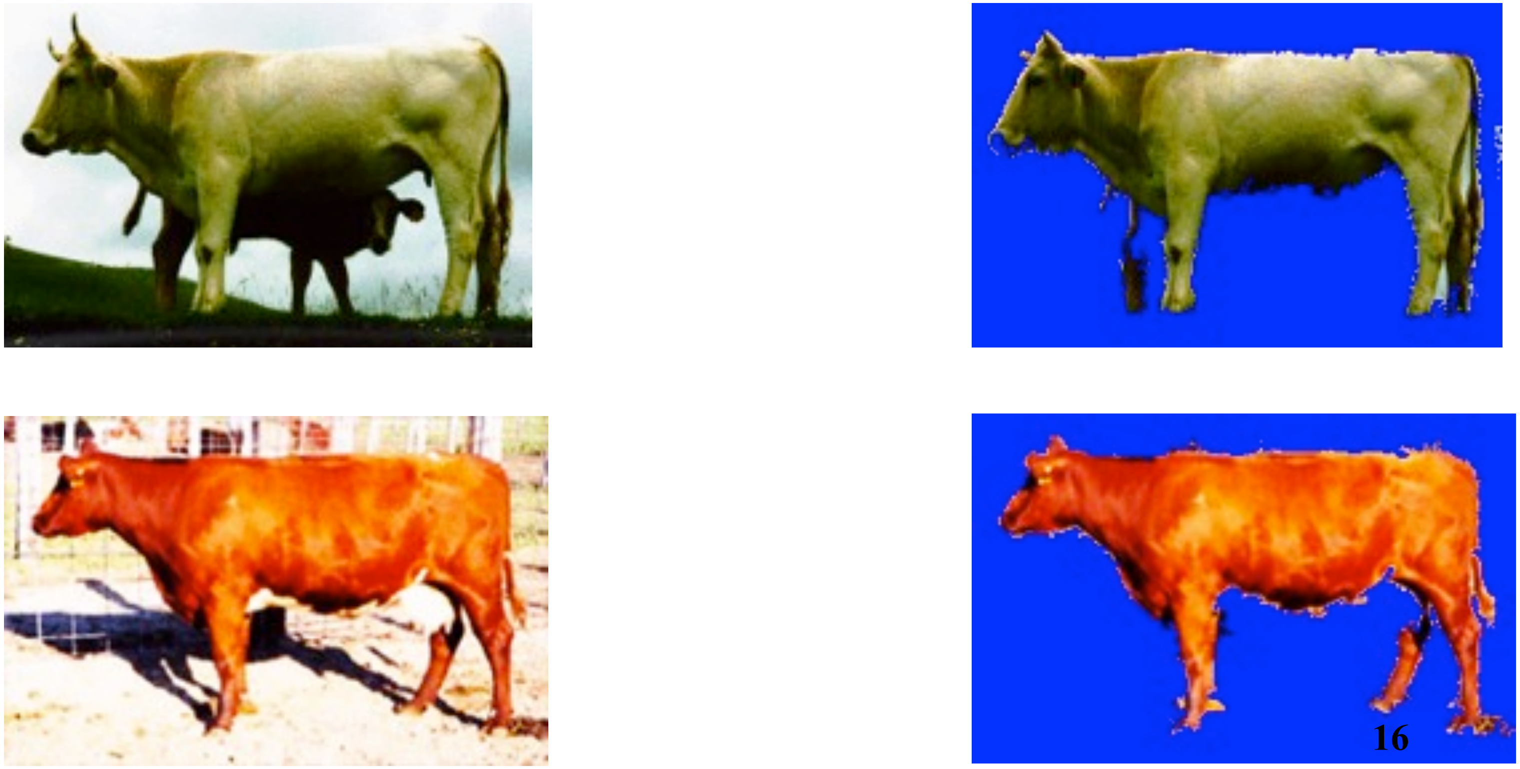
Kumar, et al. 2004,

OBJCUT: Results

Using LPS Model for Cow

In the absence of a clear boundary between object and background

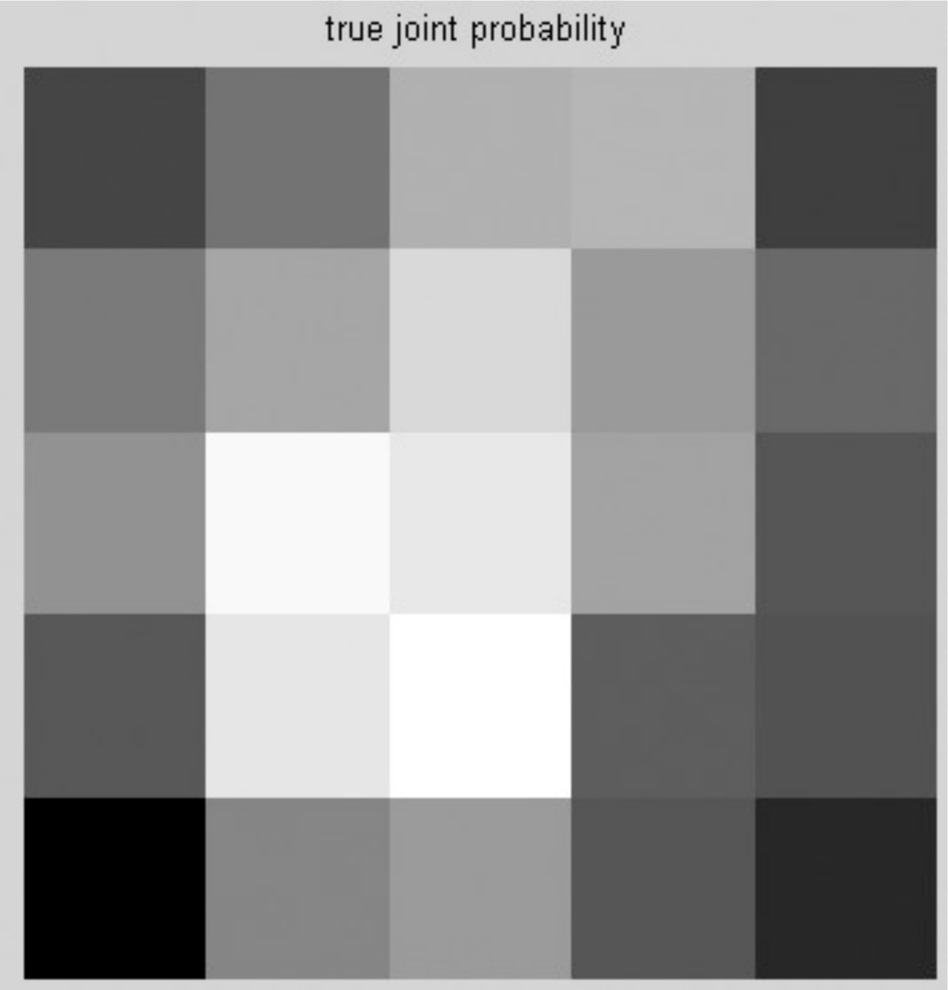
Image **Segmentation**



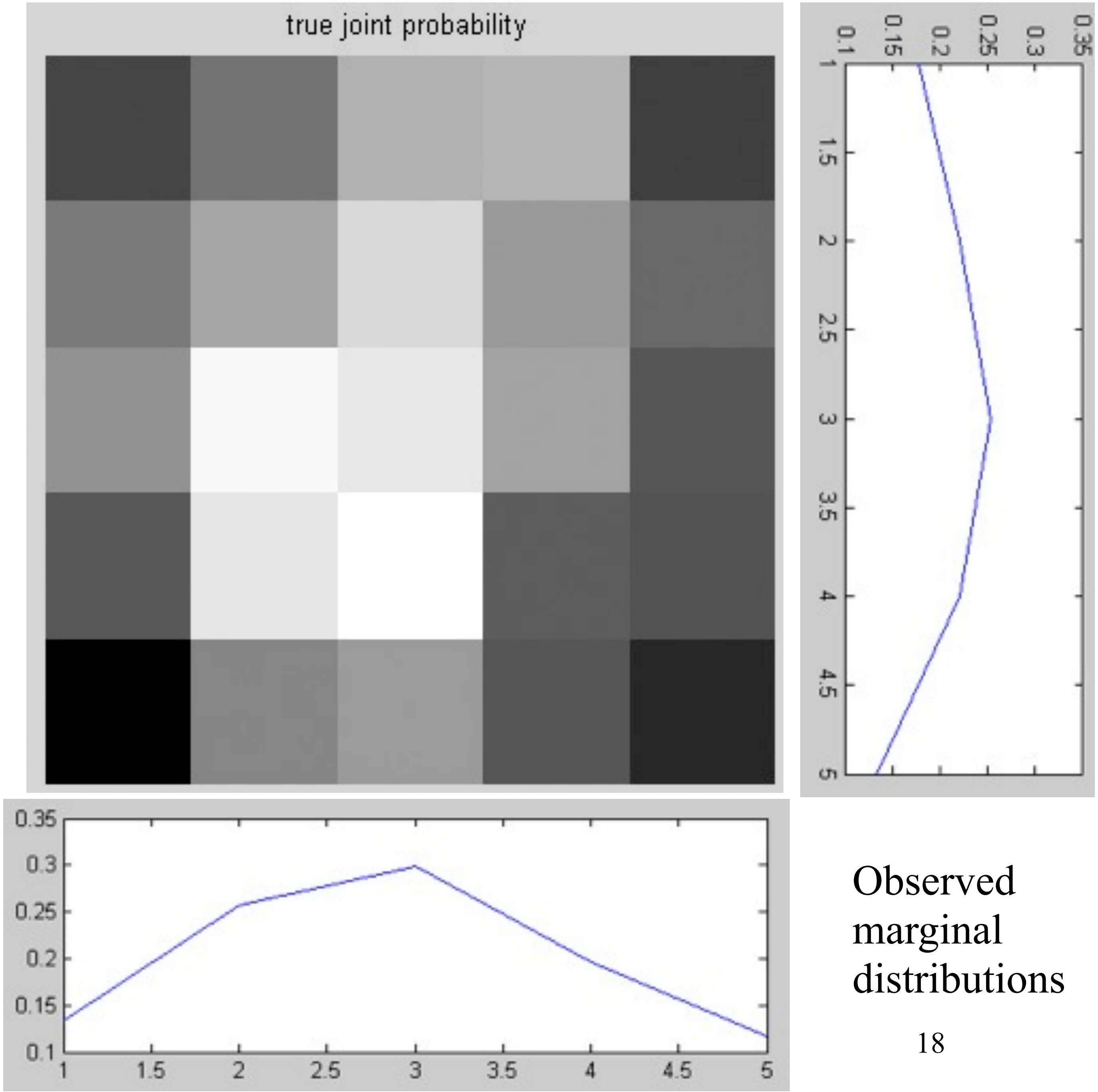
Outline of MRF section

- Inference in MRF’s.
  - – Gibbs sampling, simulated annealing
  - – Iterated conditional modes (ICM)
  - – Belief propagation
    - Application example—super-resolution
  - – Graph cuts
  - – Variational methods
- Learning MRF parameters.
  - – Iterative proportional fitting (IPF)

True joint probability

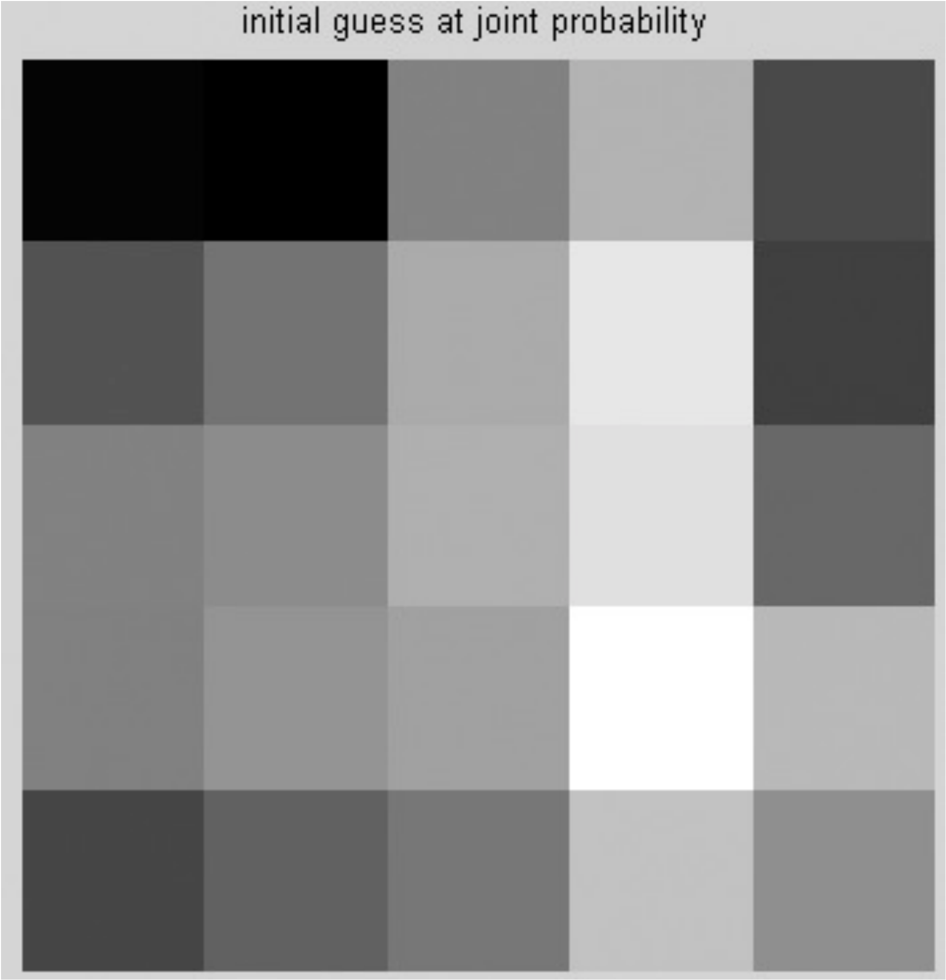


True joint probability



Observed marginal distributions

Initial guess at joint probability



IPF update equation, for maximum likelihood estimate of joint probability

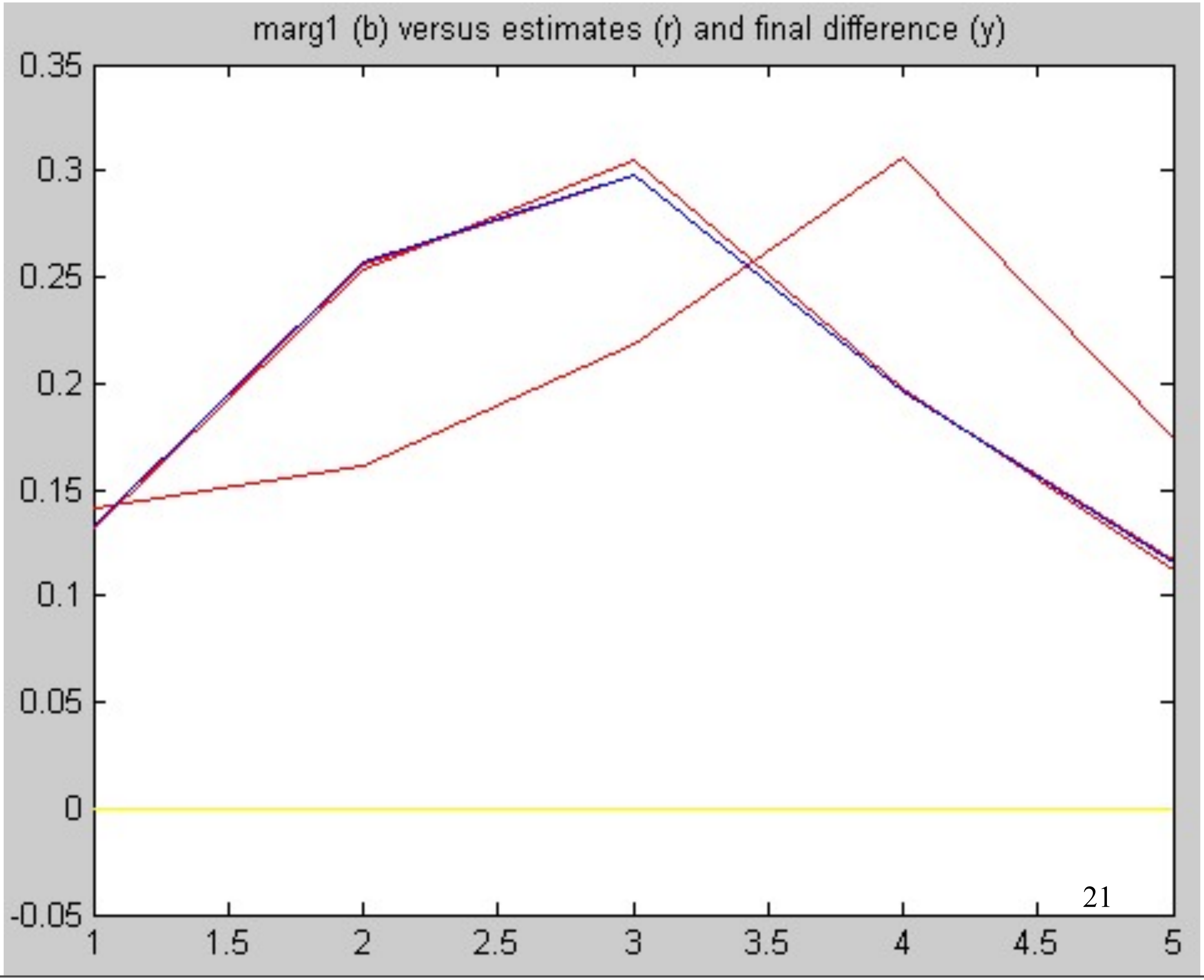
$$P(x_1, x_2, \ldots, x_d)^{(t+1)} = P(x_1, x_2, \ldots, x_d)^{(t)} \frac{P(x_i)^{\text{observed}}}{P(x_i)^{(t)}}$$

Scale the previous iteration's estimate for the joint probability by the ratio of the true to the predicted marginals.

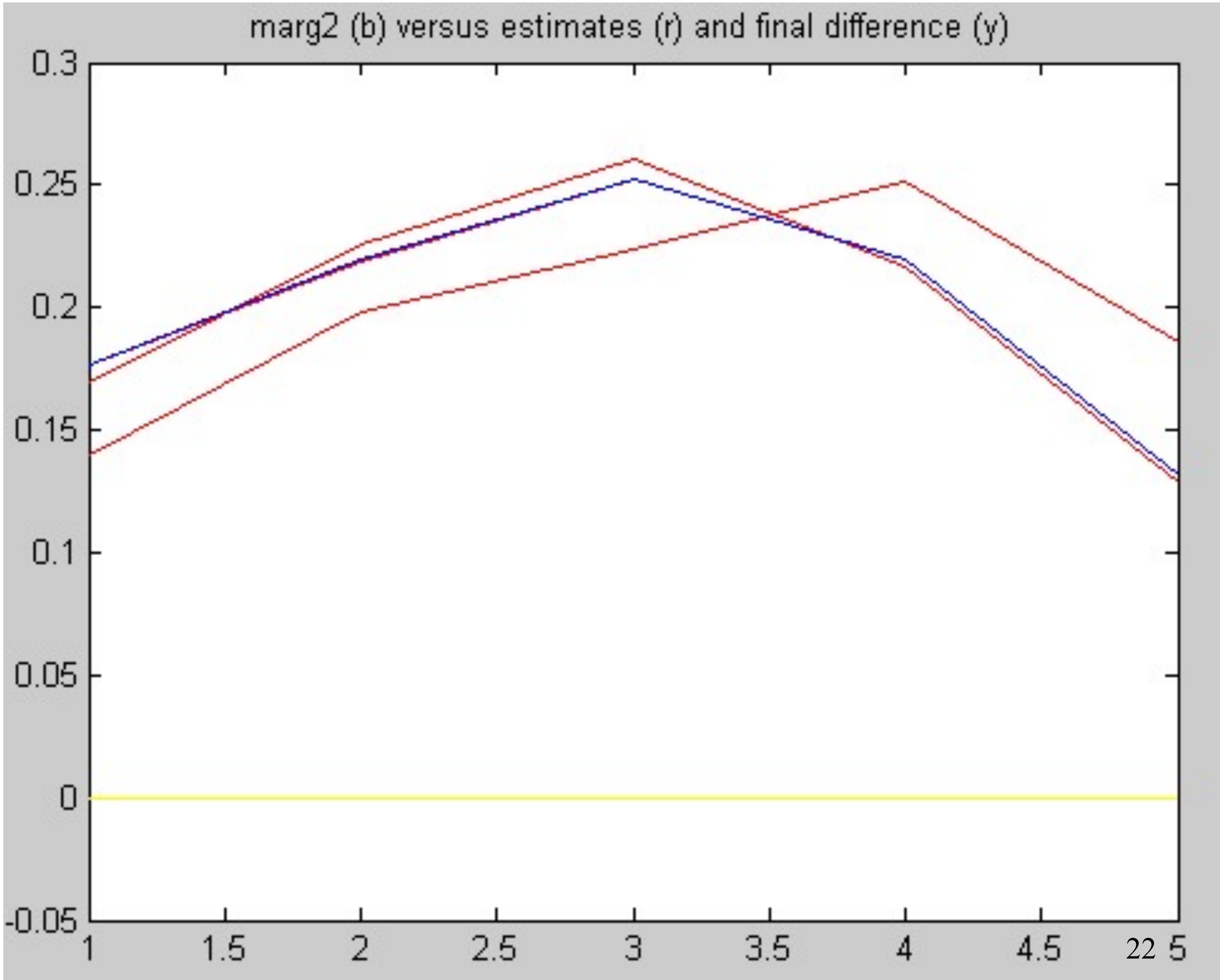
Gives gradient ascent in the likelihood of the joint probability, given the observations of the marginals.

See: Michael Jordan's book on graphical models

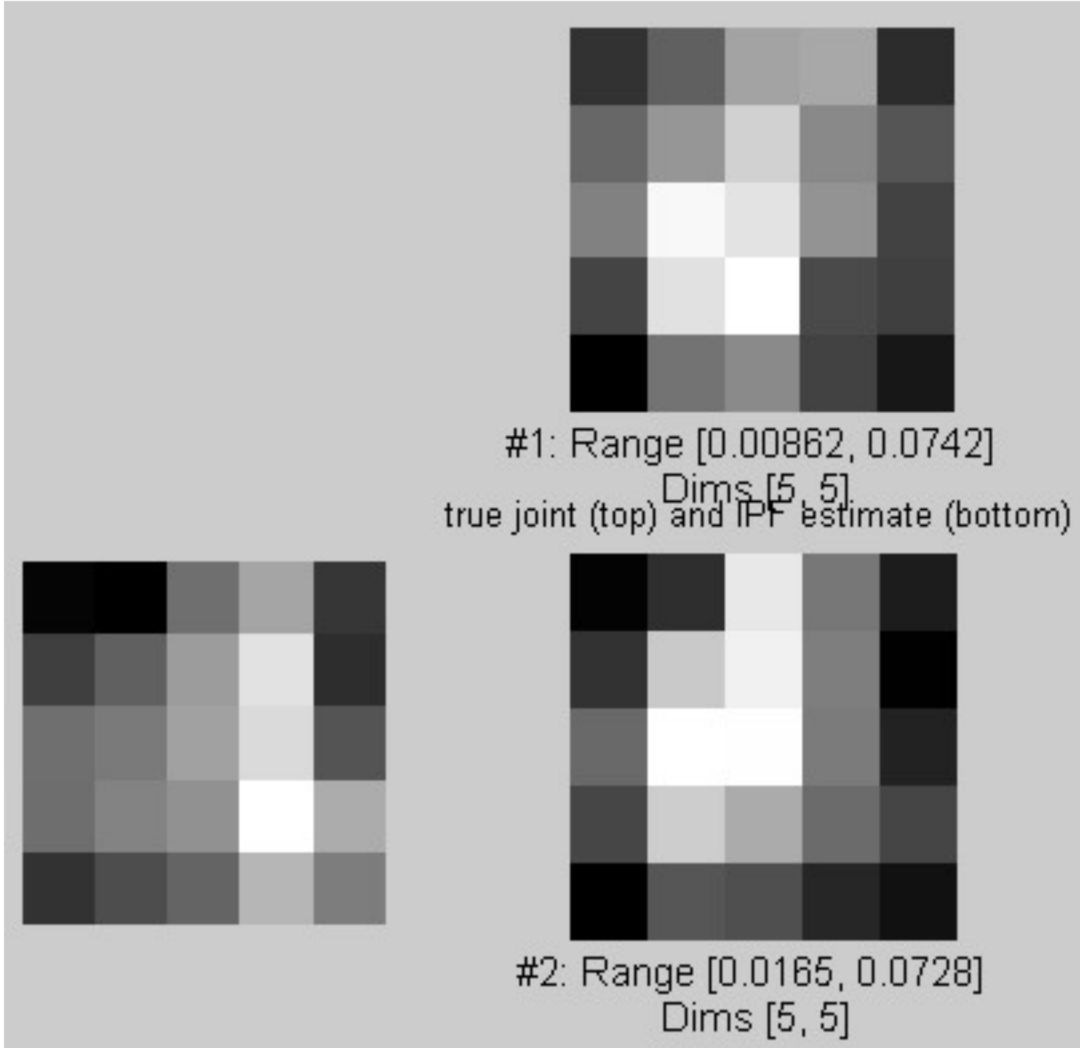
Convergence to correct marginals by IPF algorithm



Convergence of to correct marginals by IPF algorithm



IPF results for this example: comparison of joint probabilities



True joint probability

Initial guess

Final maximum entropy estimate

Application to MRF parameter estimation

- Can show that for the ML estimate of the clique potentials, $\phi_c(x_c)$, the empirical marginals equal the model marginals,

$$\tilde{p}(x_c) = p(x_c)$$

- Because the model marginals are proportional to the clique potentials, we have the IPF update rule for $\phi_c(x_c)$, which scales the model marginals to equal the observed marginals:

$$\phi_C^{(t+1)}(x_c) = \phi_c^{(t)}(x_c)\frac{\tilde{p}(x_c)}{p^{(t)}(x_c)}$$

- Performs coordinate ascent in the likelihood of the MRF parameters, given the observed data.

Reference: unpublished notes by Michael Jordan, and by Roweis: http://www.cs.toronto.edu/~roweis/csc412-2004/notes/lec11x.pdf

Learning MRF parameters, labeled data

Iterative proportional fitting lets you make a maximum likelihood estimate a joint distribution from observations of various marginal distributions.

Applied to learning MRF clique potentials:

(1) measure the pairwise marginal statistics (histogram state co-occurrences in labeled training data).

(2) guess the clique potentials (use measured marginals to start).

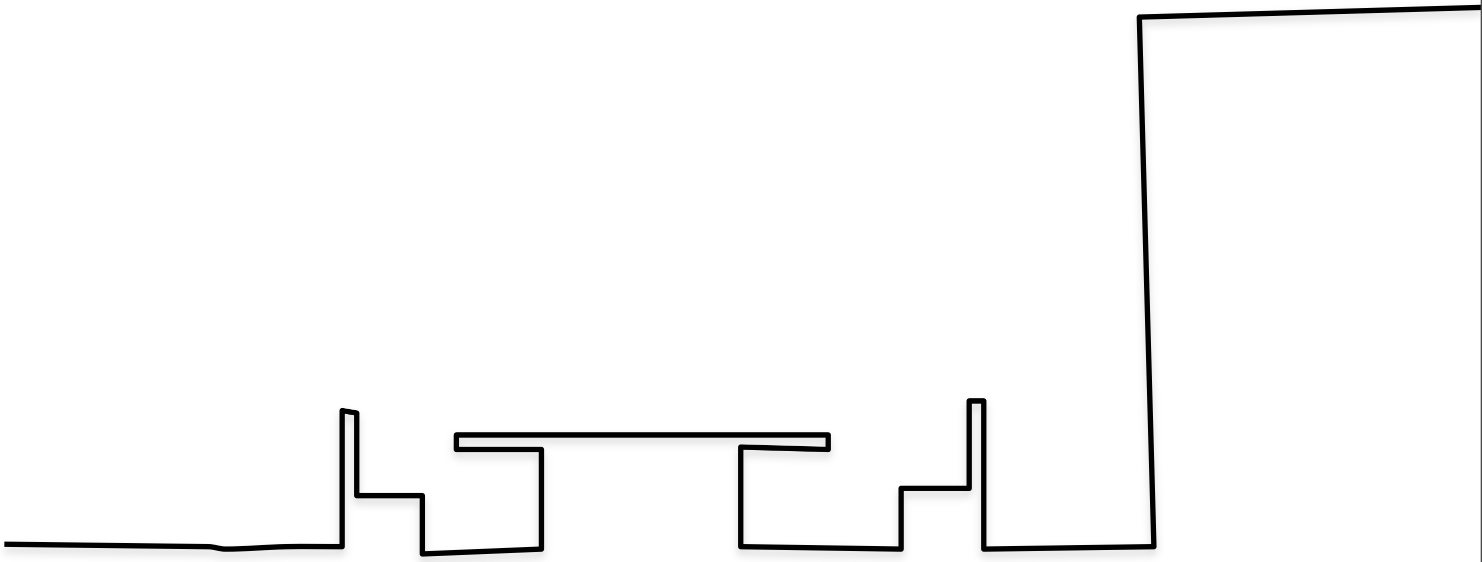
(3) do inference to calculate the model's marginals for every node pair.

(4) scale each clique potential for each state pair by the empirical over model marginal

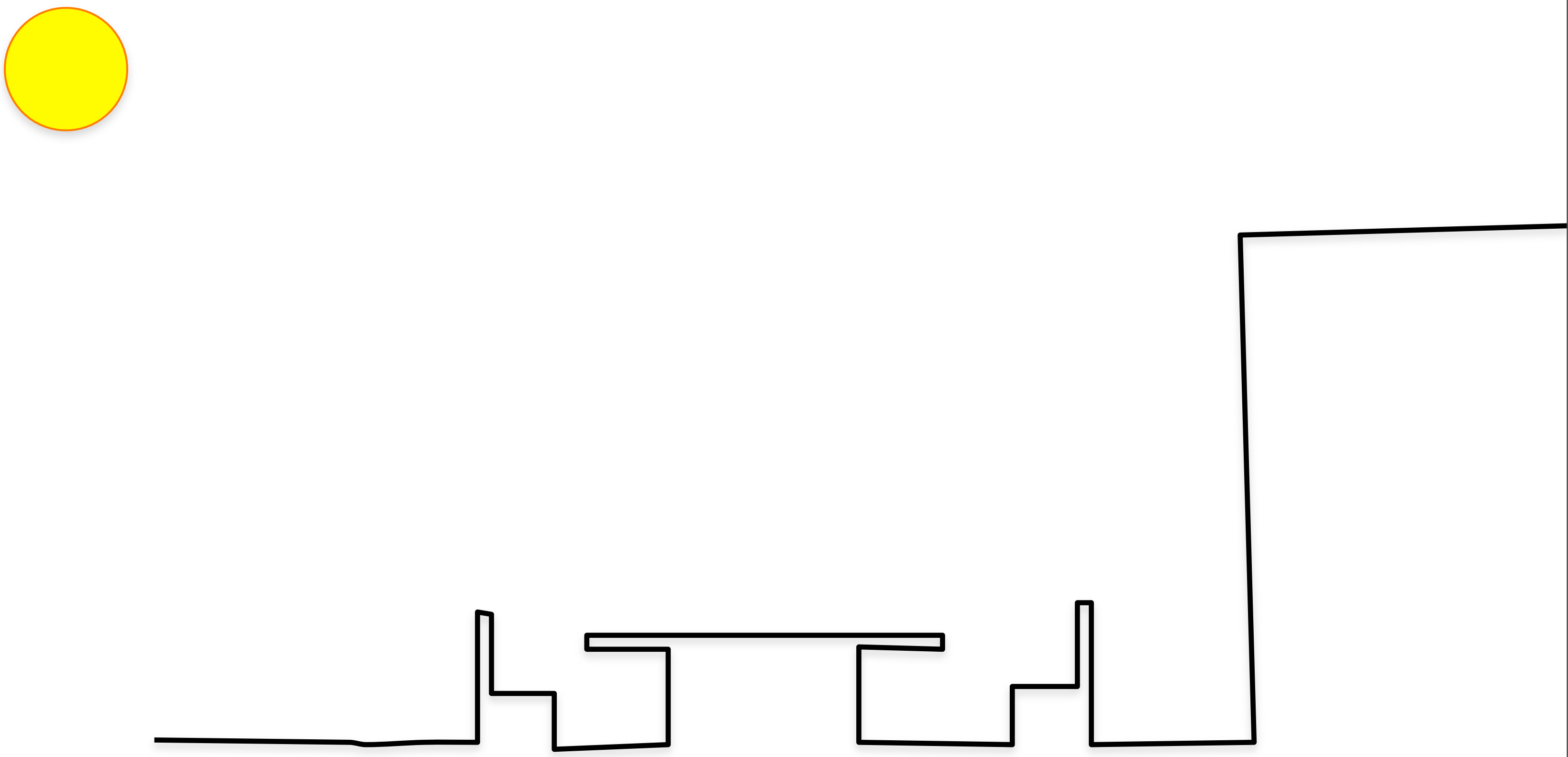
Image formation

The structure of ambient light

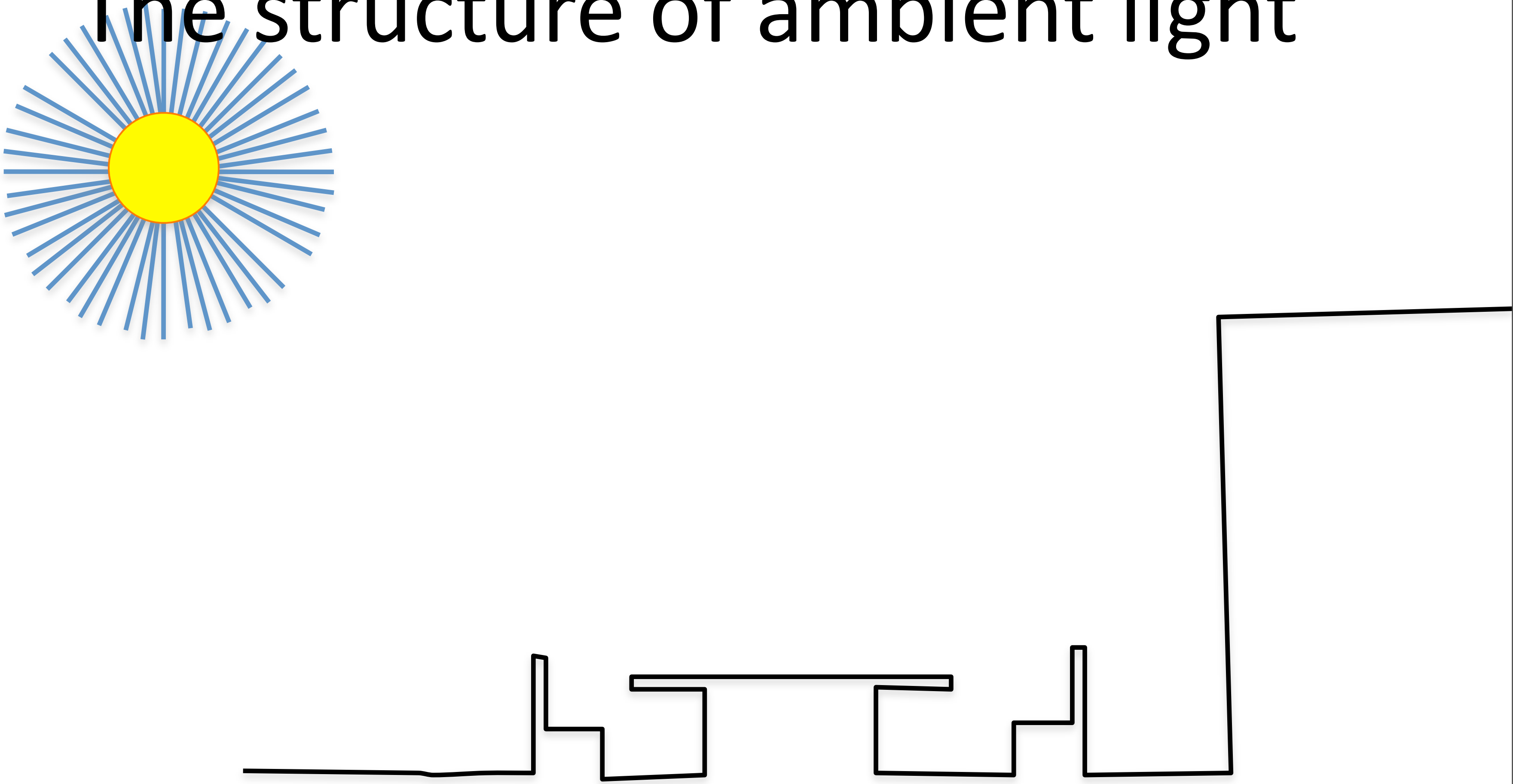
The structure of ambient light



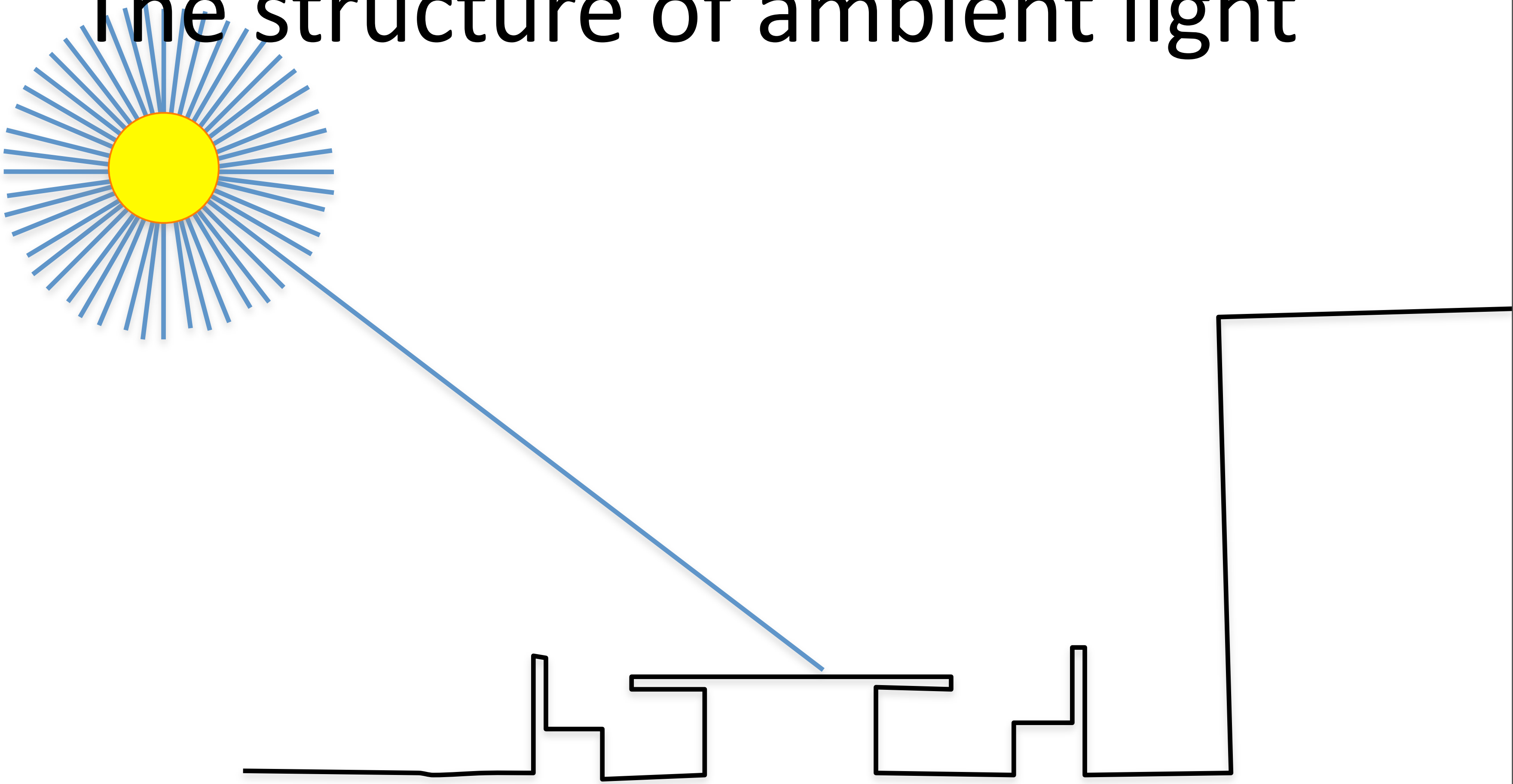
The structure of ambient light



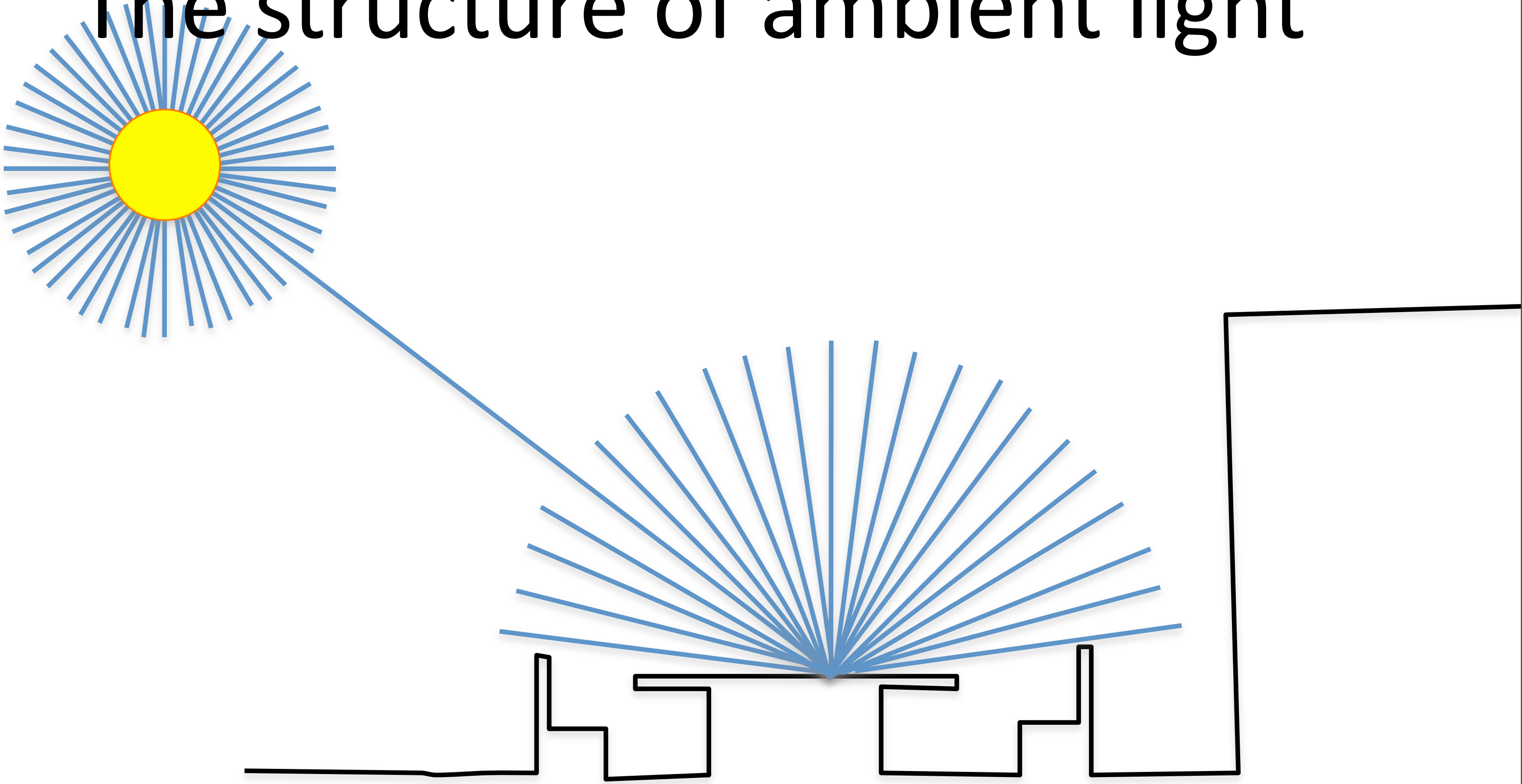
The structure of ambient light



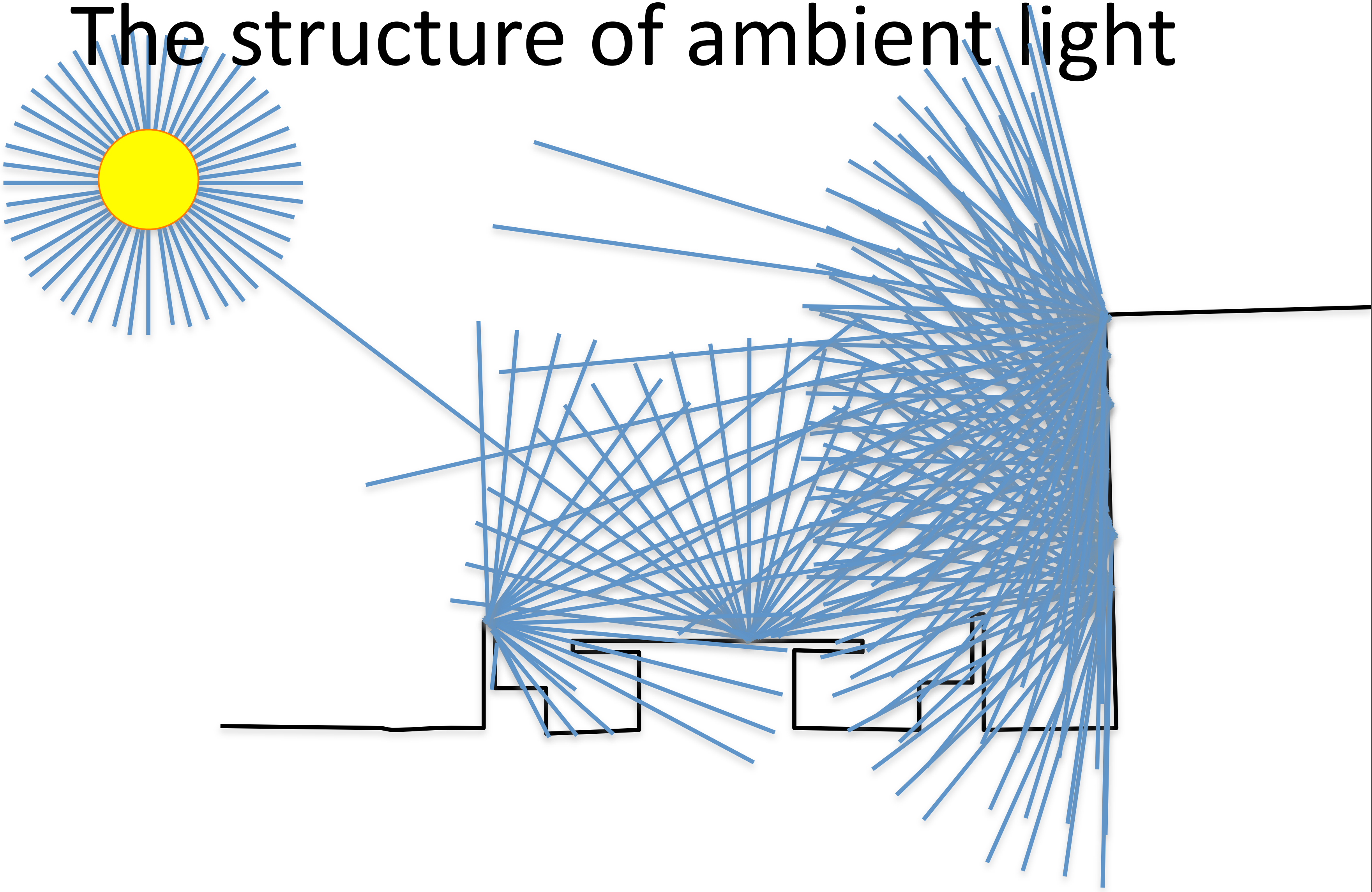
The structure of ambient light



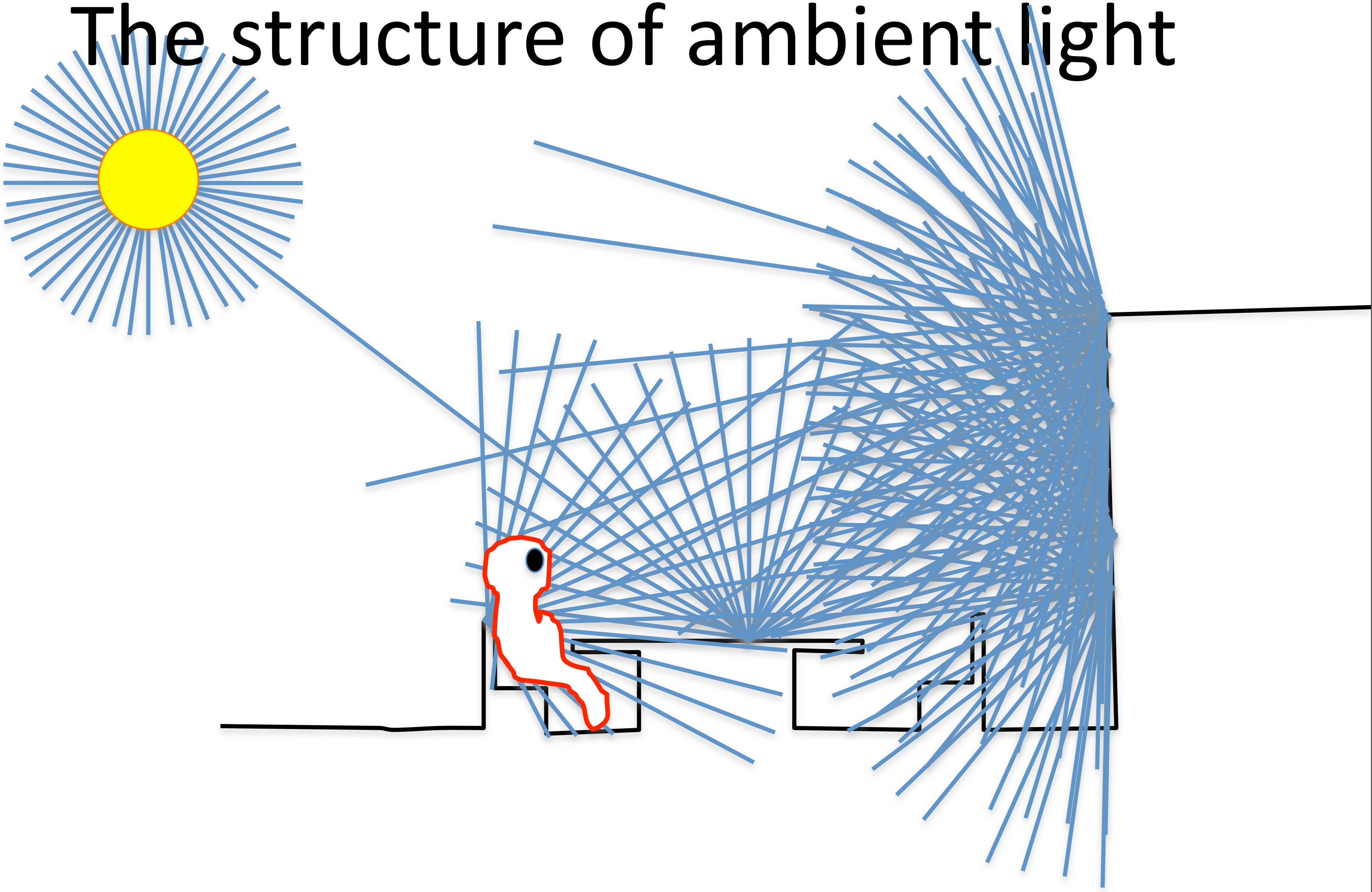
The structure of ambient light



The structure of ambient light

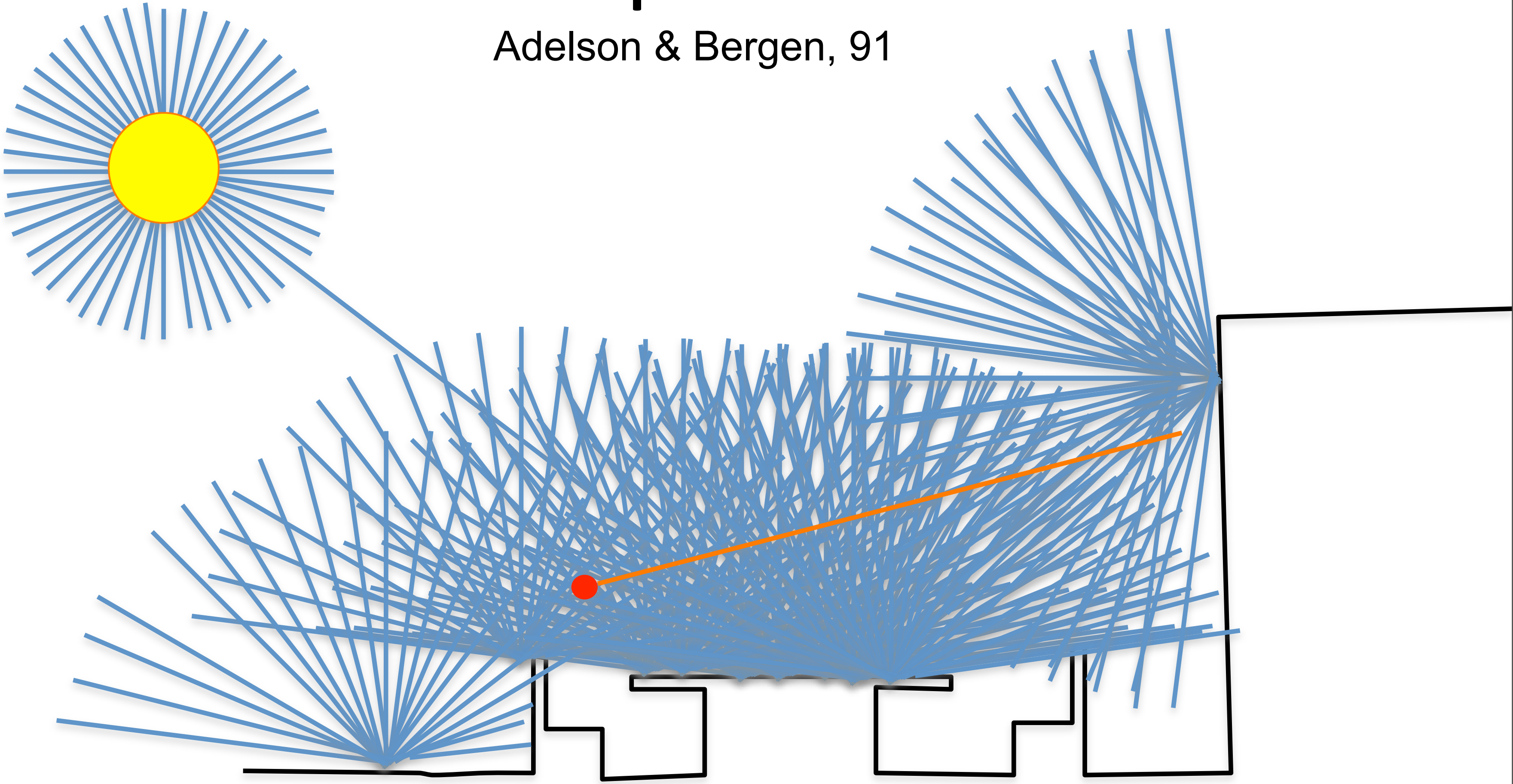


The structure of ambient light

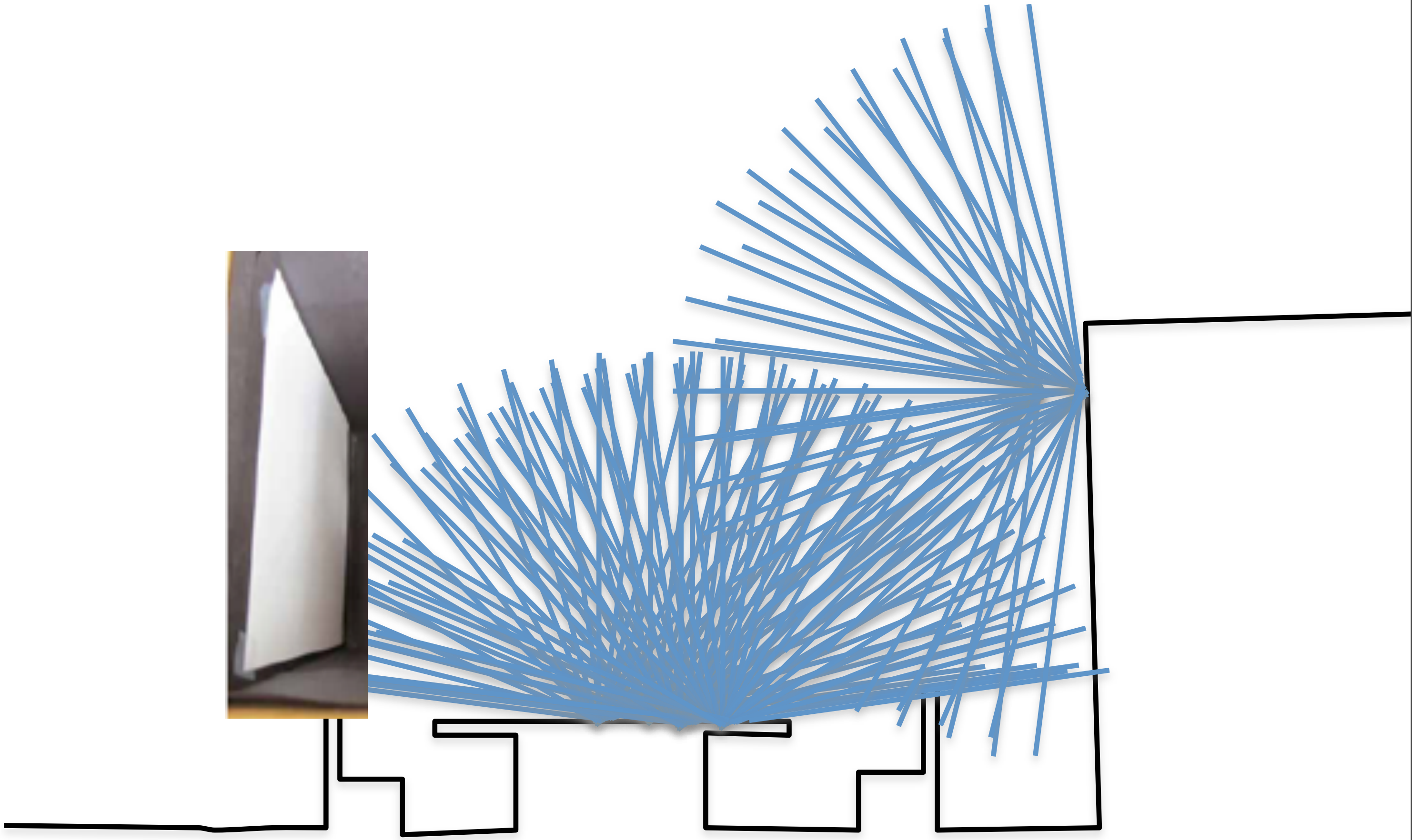


The Plenoptic Function

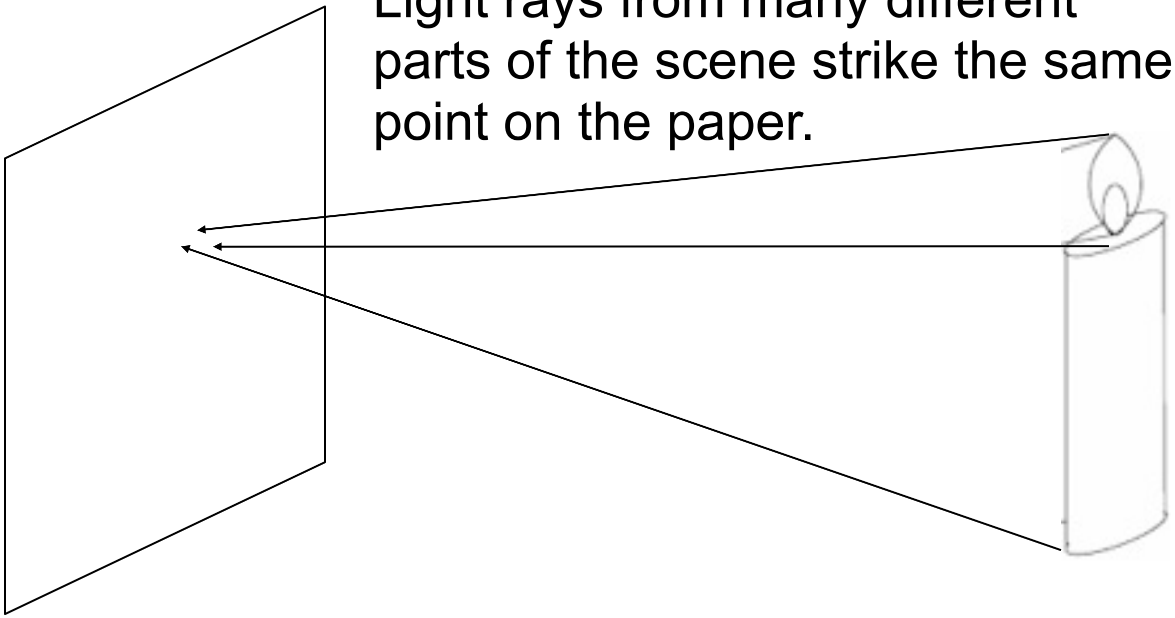
Adelson & Bergen, 91



Measuring the Plenoptic function



Why is there no picture appearing on the paper?

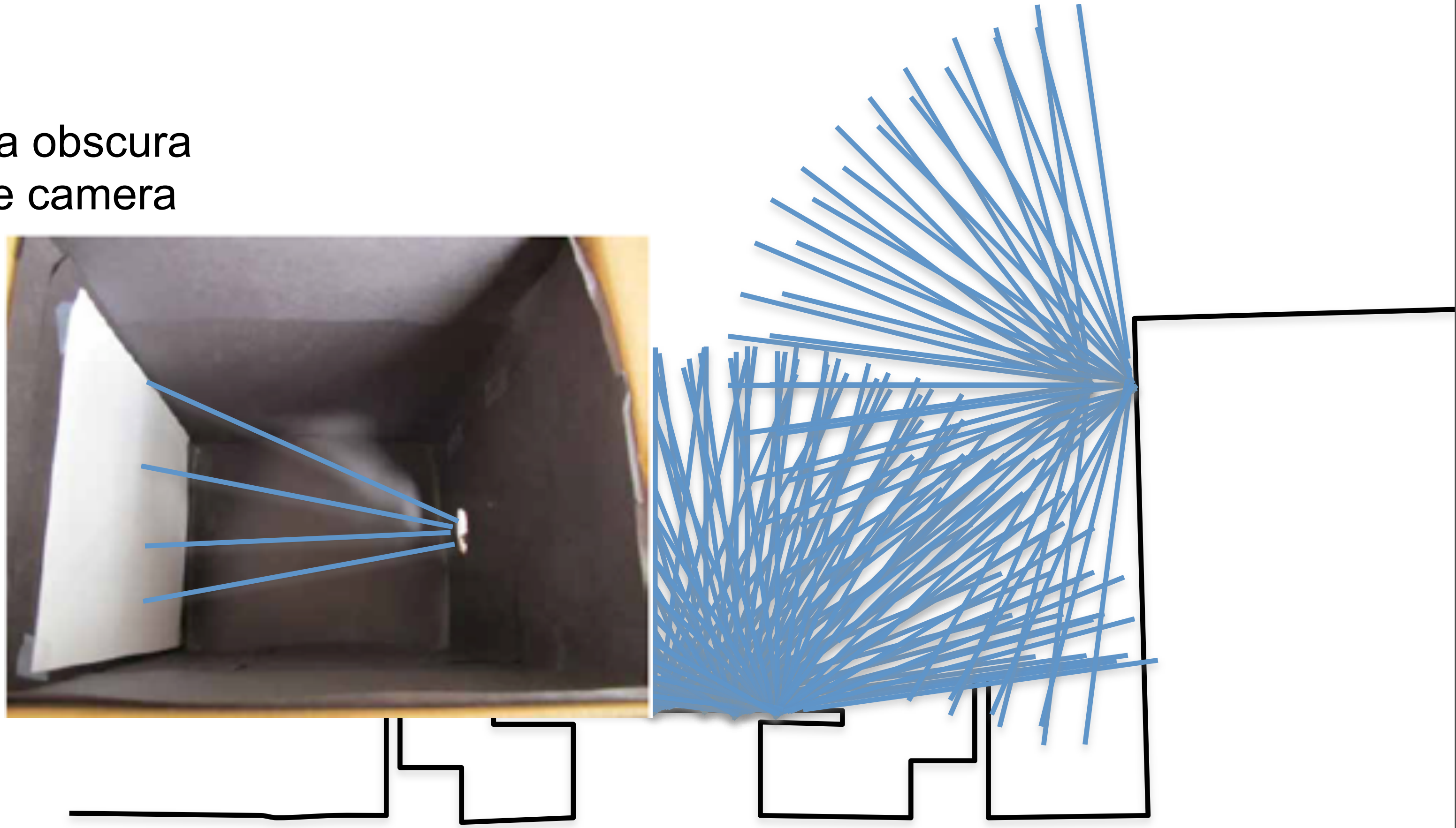


Light rays from many different parts of the scene strike the same point on the paper.

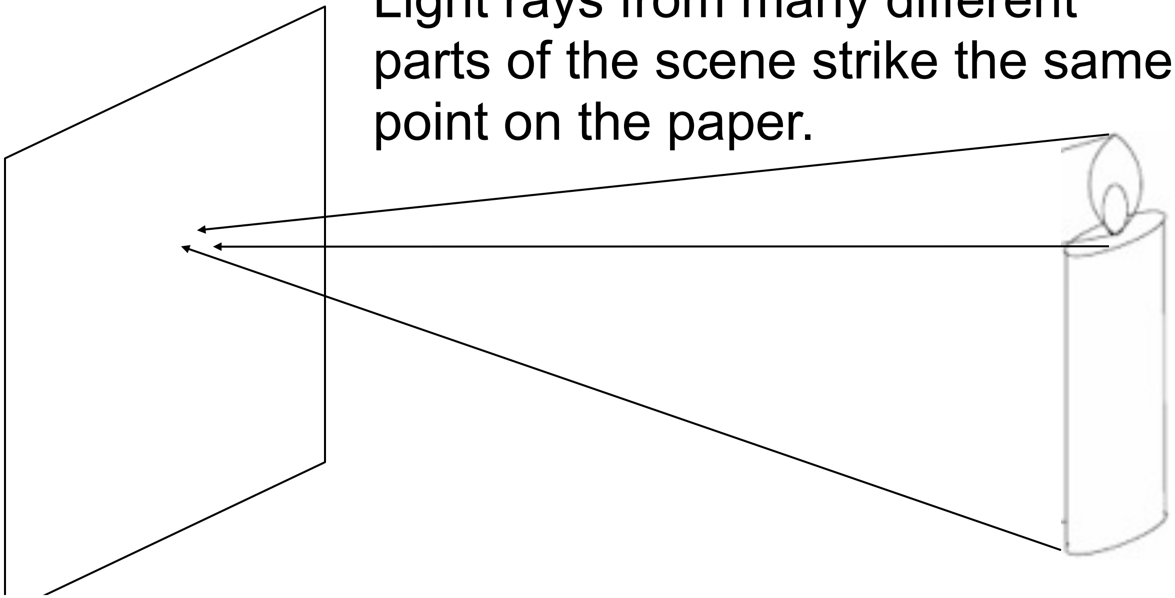
Forsyth & Ponce

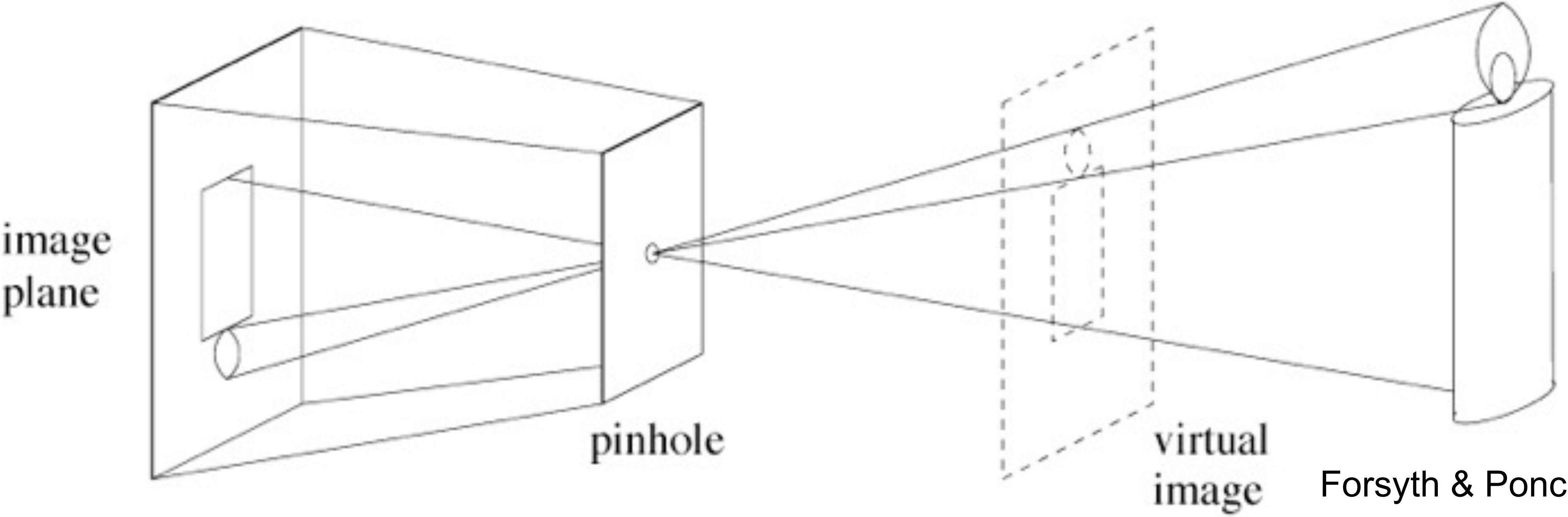
Measuring the Plenoptic function

The camera obscura
The pinhole camera



Light rays from many different parts of the scene strike the same point on the paper.



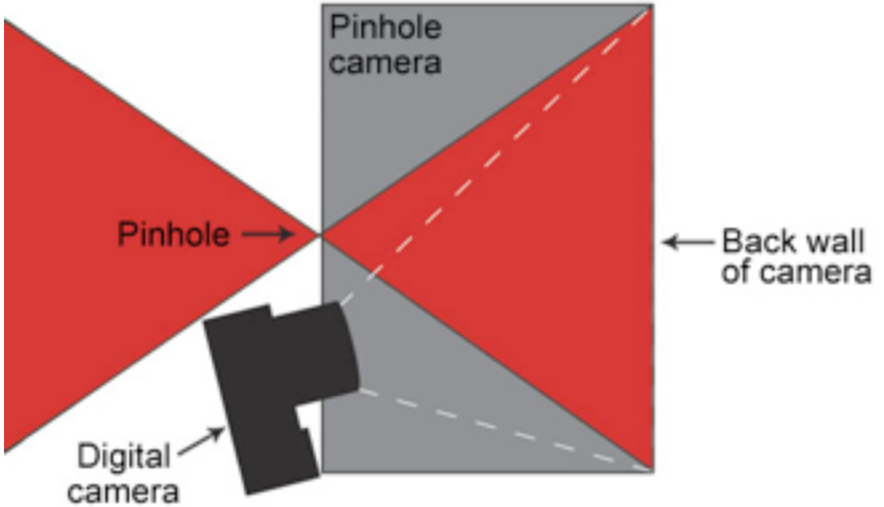


Forsyth & Ponce

The pinhole camera only allows rays from one point in the scene to strike each point of the paper.

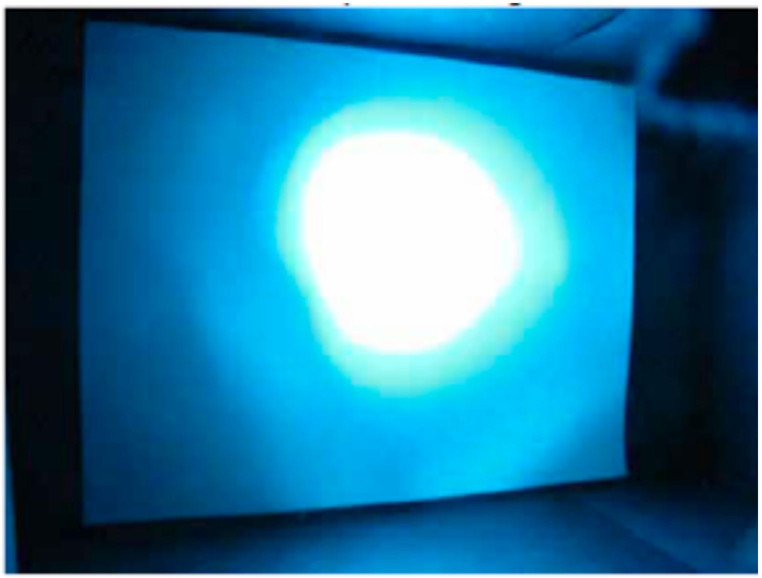
pinhole camera demos

Problem Set 7

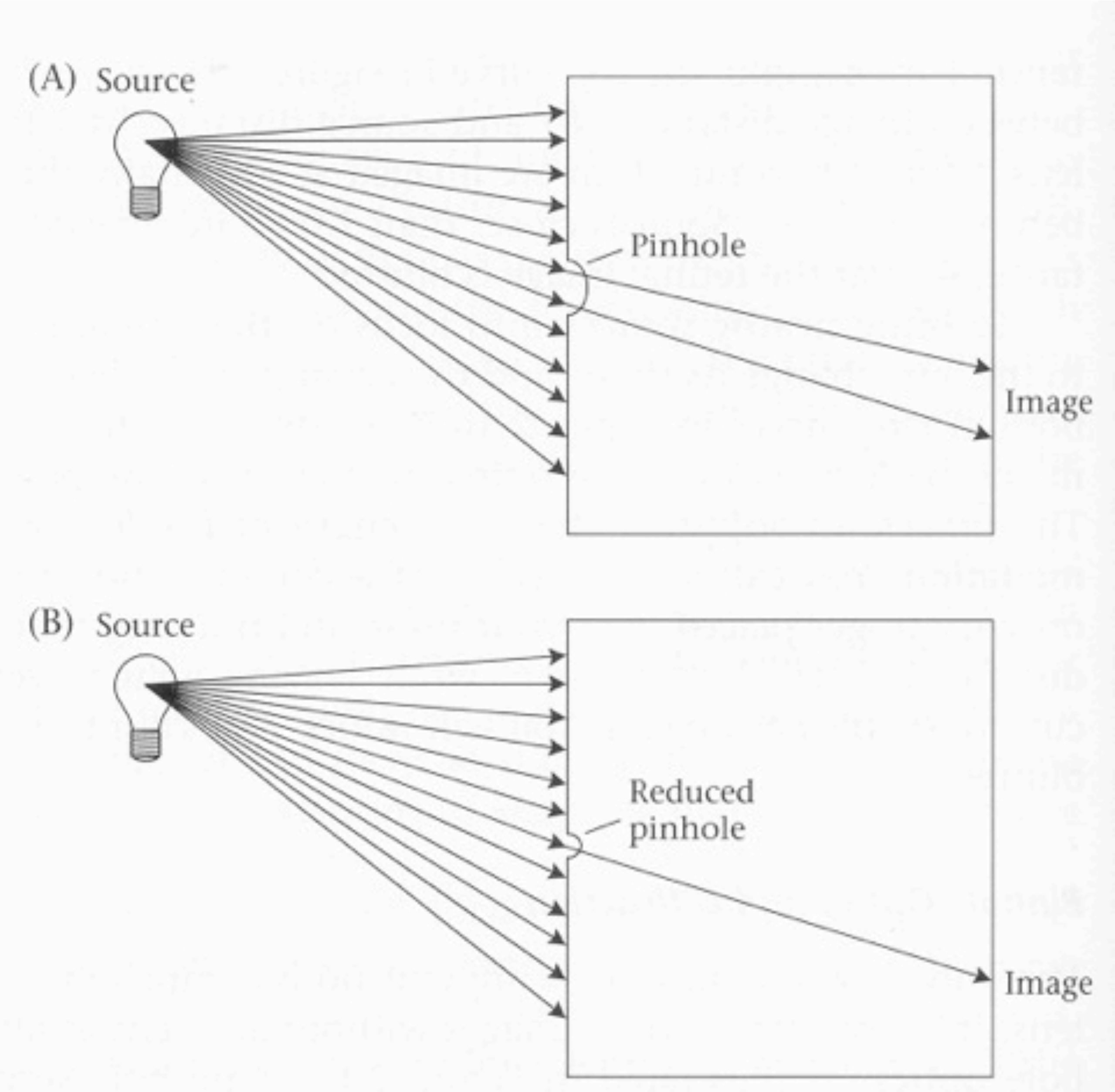


http://www.foundphotography.com/PhotoThoughts/archives/2005/04/pinhole_camera_2.html

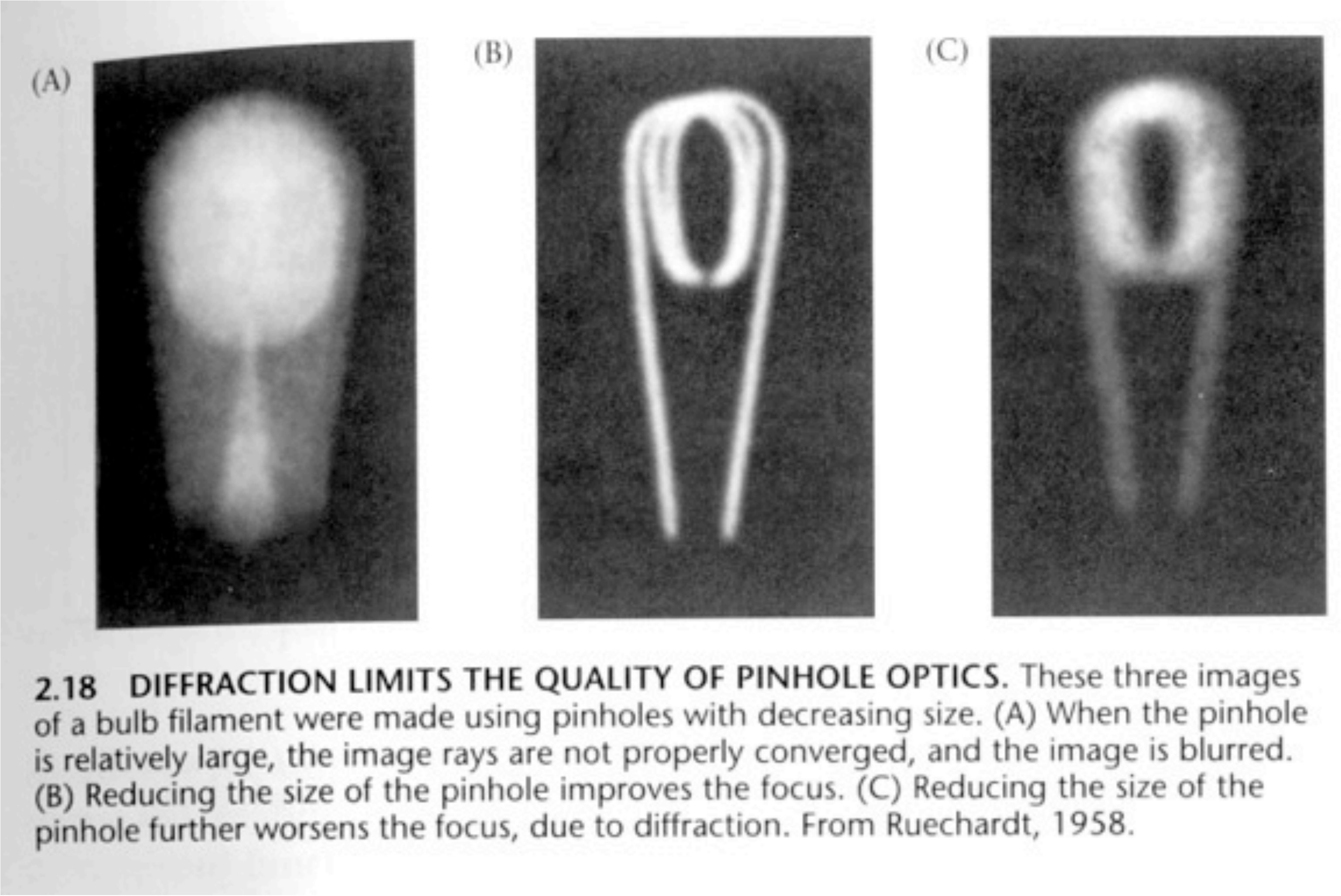
Problem Set 7



Effect of pinhole size



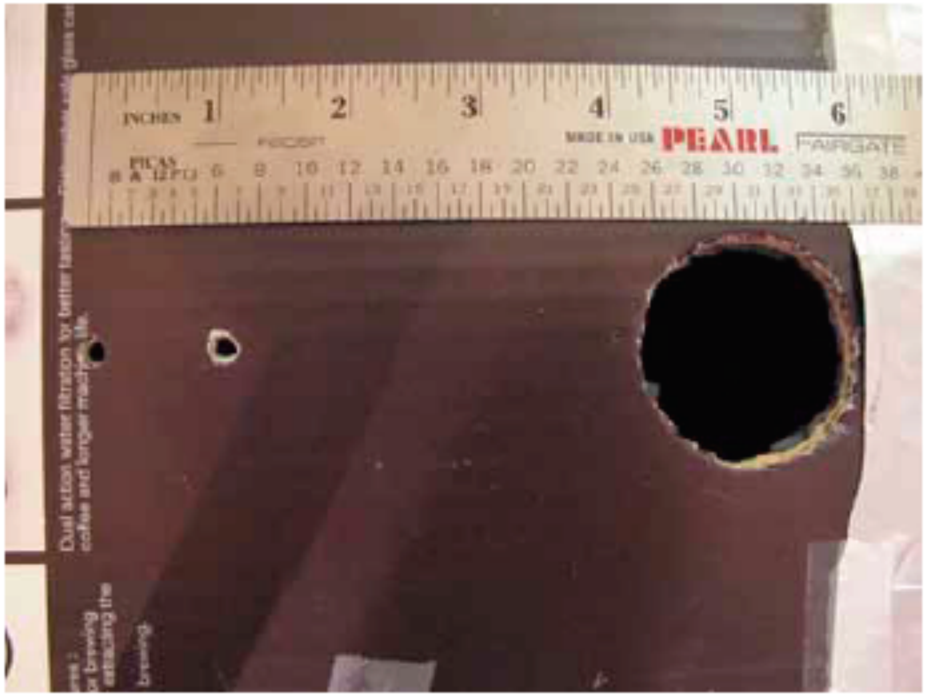
Wandell, Foundations of Vision, Sinauer, 1995



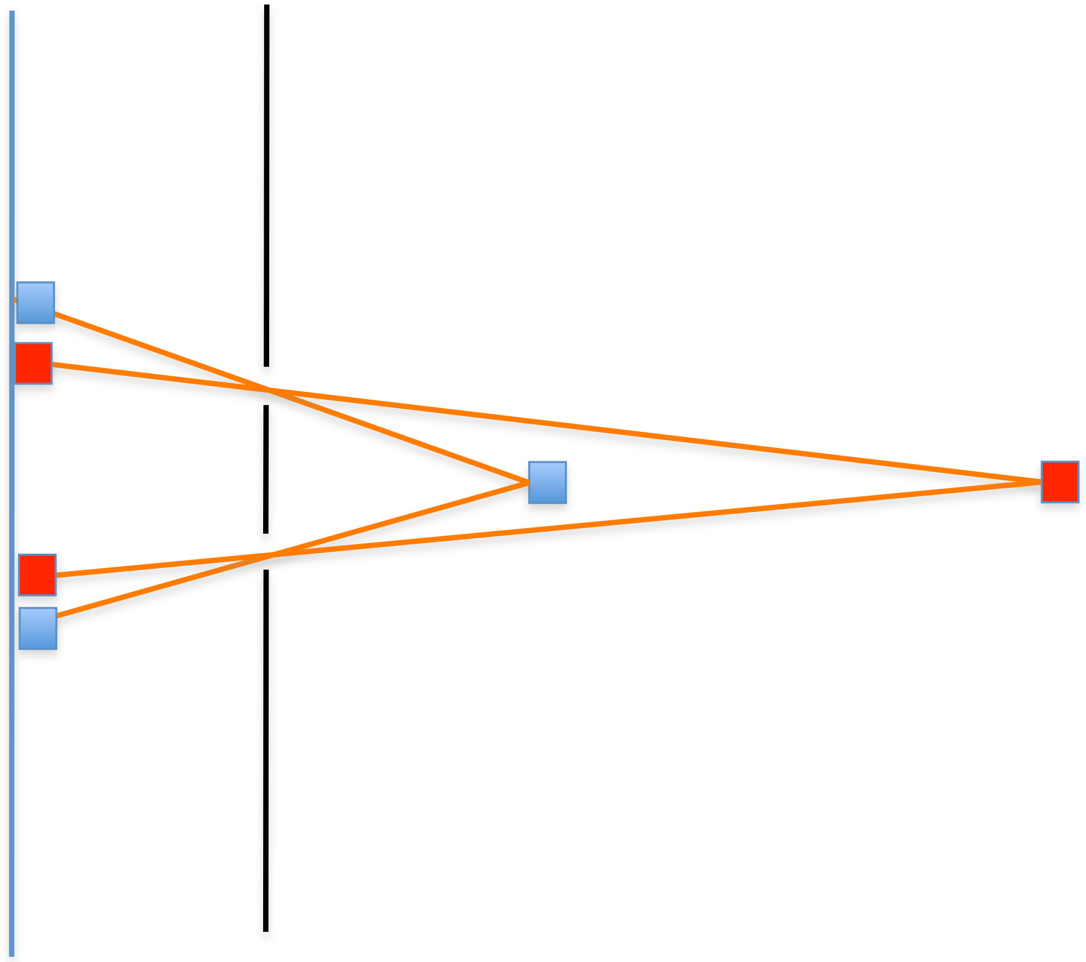
2.18 DIFFRACTION LIMITS THE QUALITY OF PINHOLE OPTICS. These three images of a bulb filament were made using pinholes with decreasing size. (A) When the pinhole is relatively large, the image rays are not properly converged, and the image is blurred. (B) Reducing the size of the pinhole improves the focus. (C) Reducing the size of the pinhole further worsens the focus, due to diffraction. From Ruechardt, 1958.

Wandell, Foundations of Vision, Sinauer, 1995

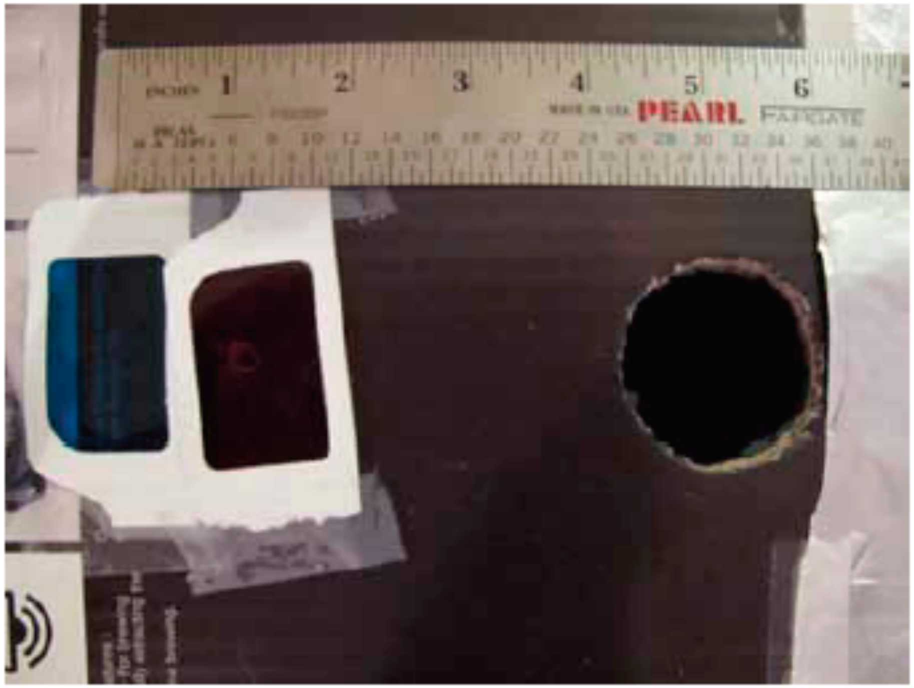
Playing with pinholes



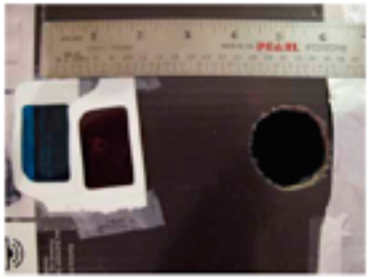
Two pinholes



Anaglyph pinhole camera

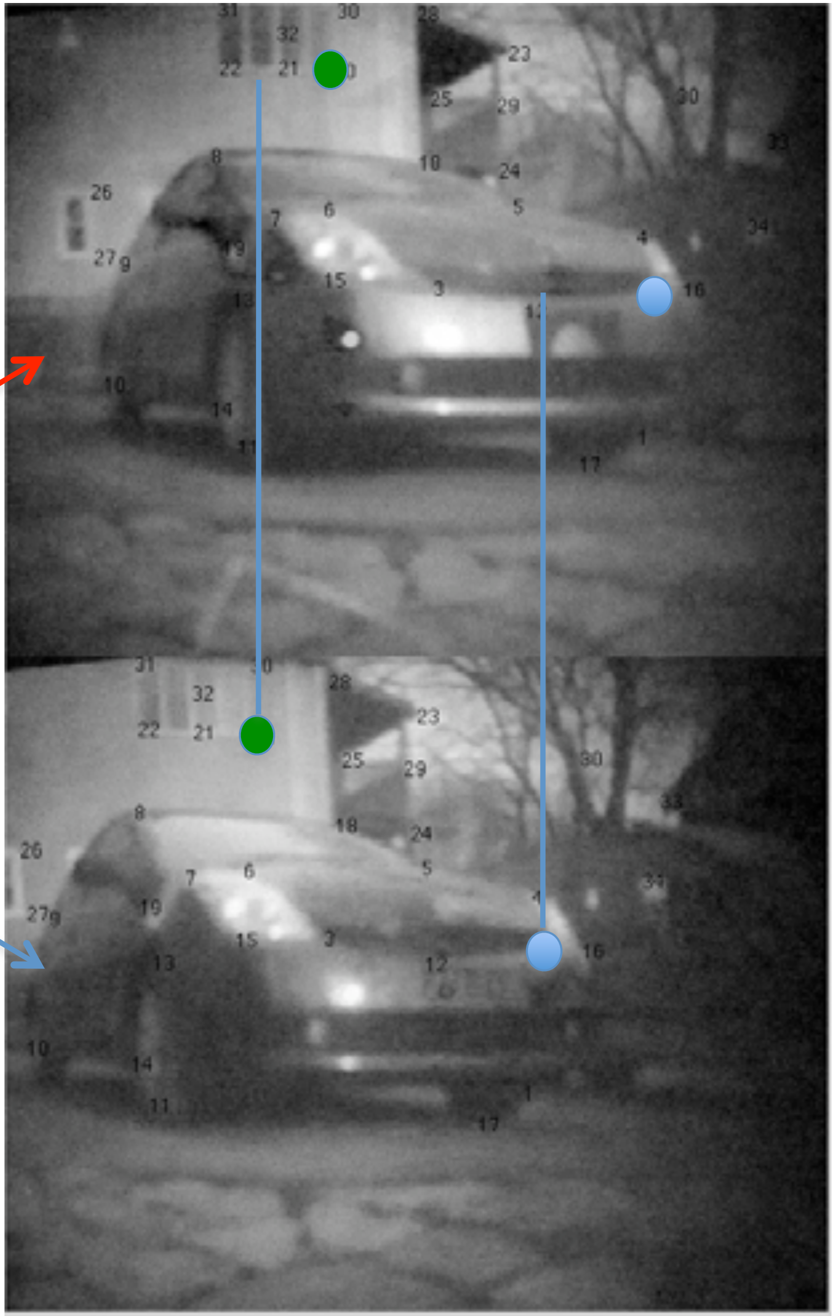


Anaglyph pinhole camera

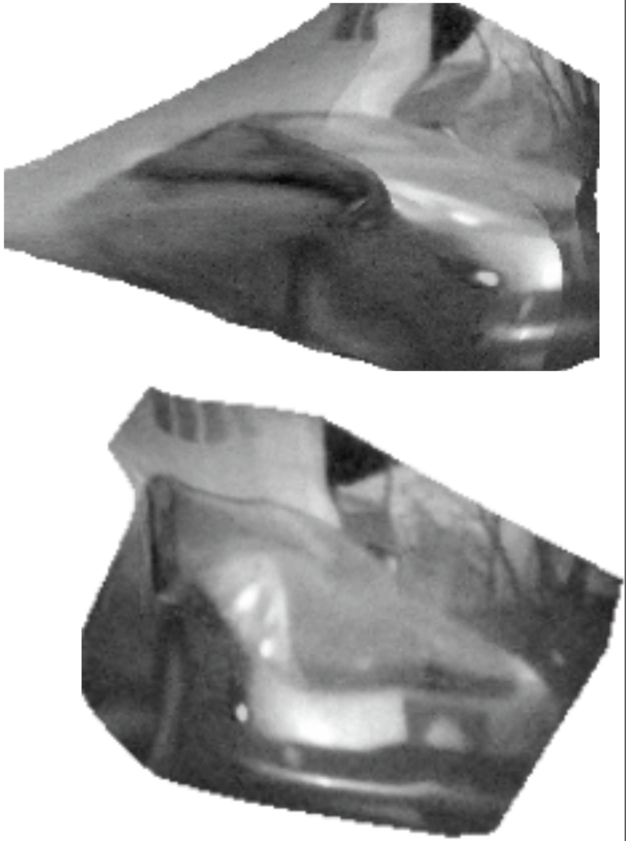


Anaglyph pinhole camera

Anaglyph



Synthesis of new views



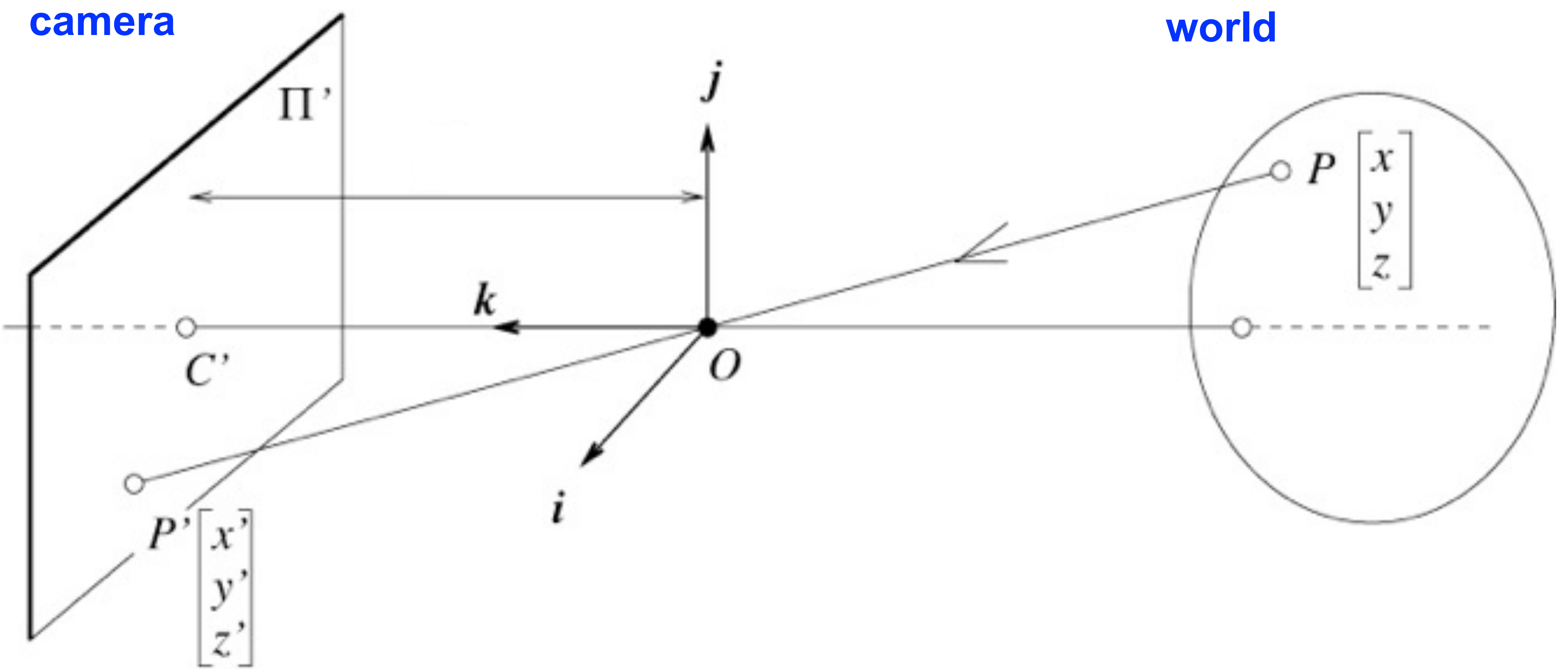
Problem set 7

- Build the device
- Take some pictures and put them in the report
- Take anaglyph images
- Calibrate camera
- Recover depth for some points in the image

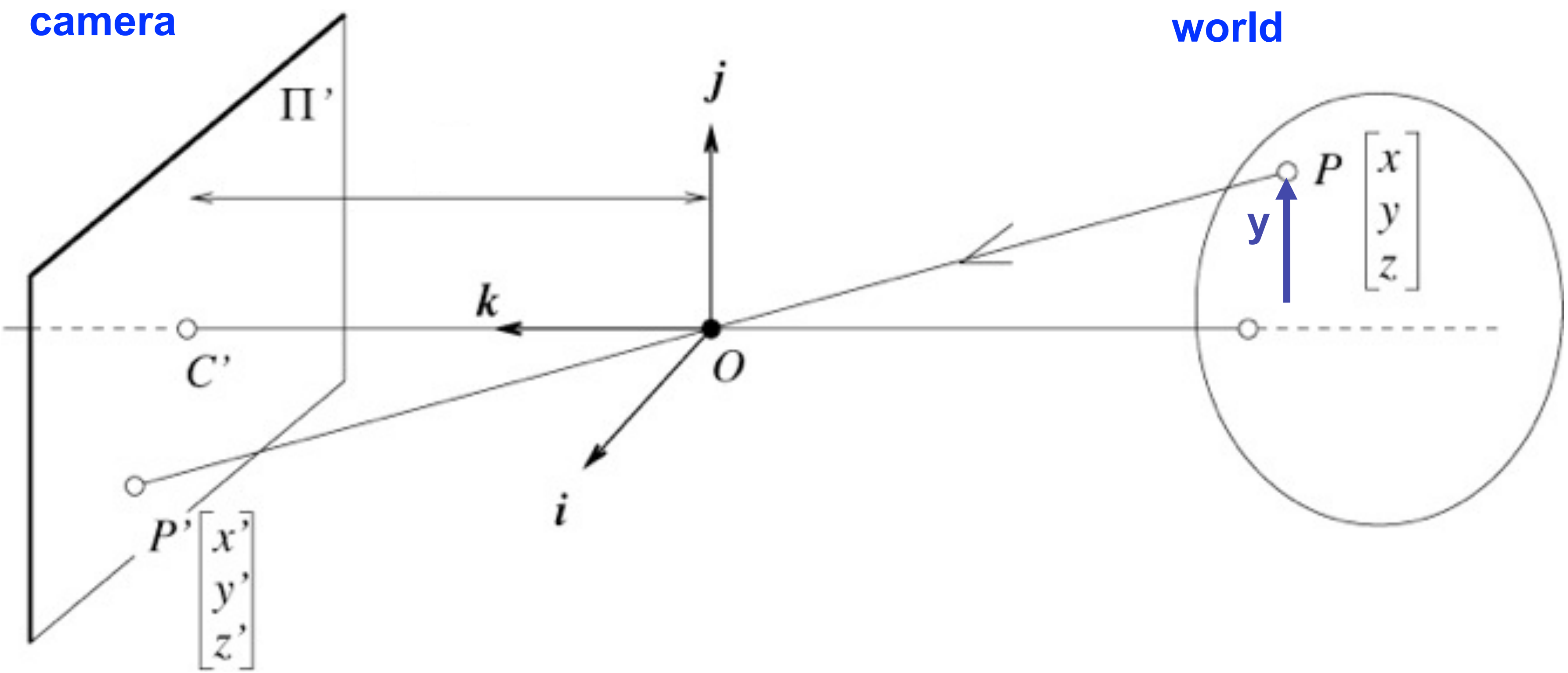
Cameras, lenses, and calibration

- Camera models
- Projections
- Calibration
- Lenses

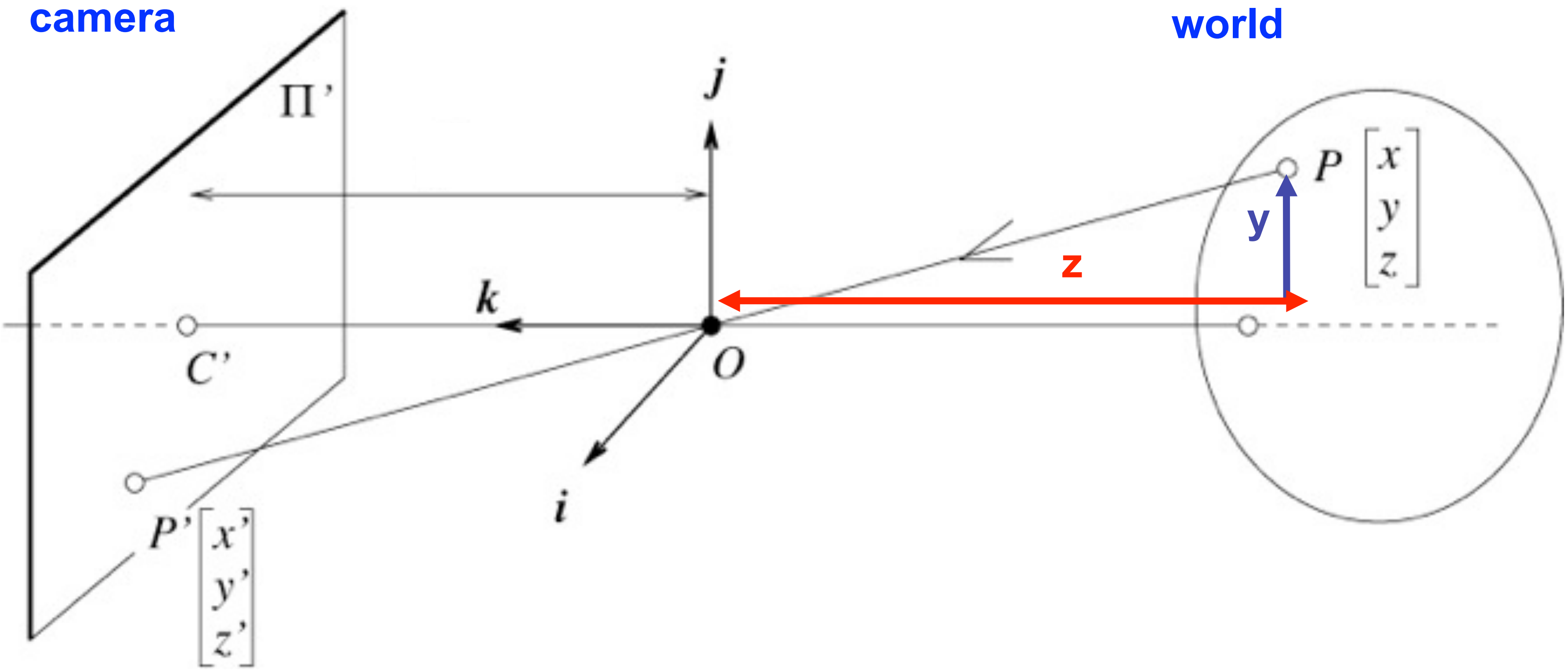
Perspective projection



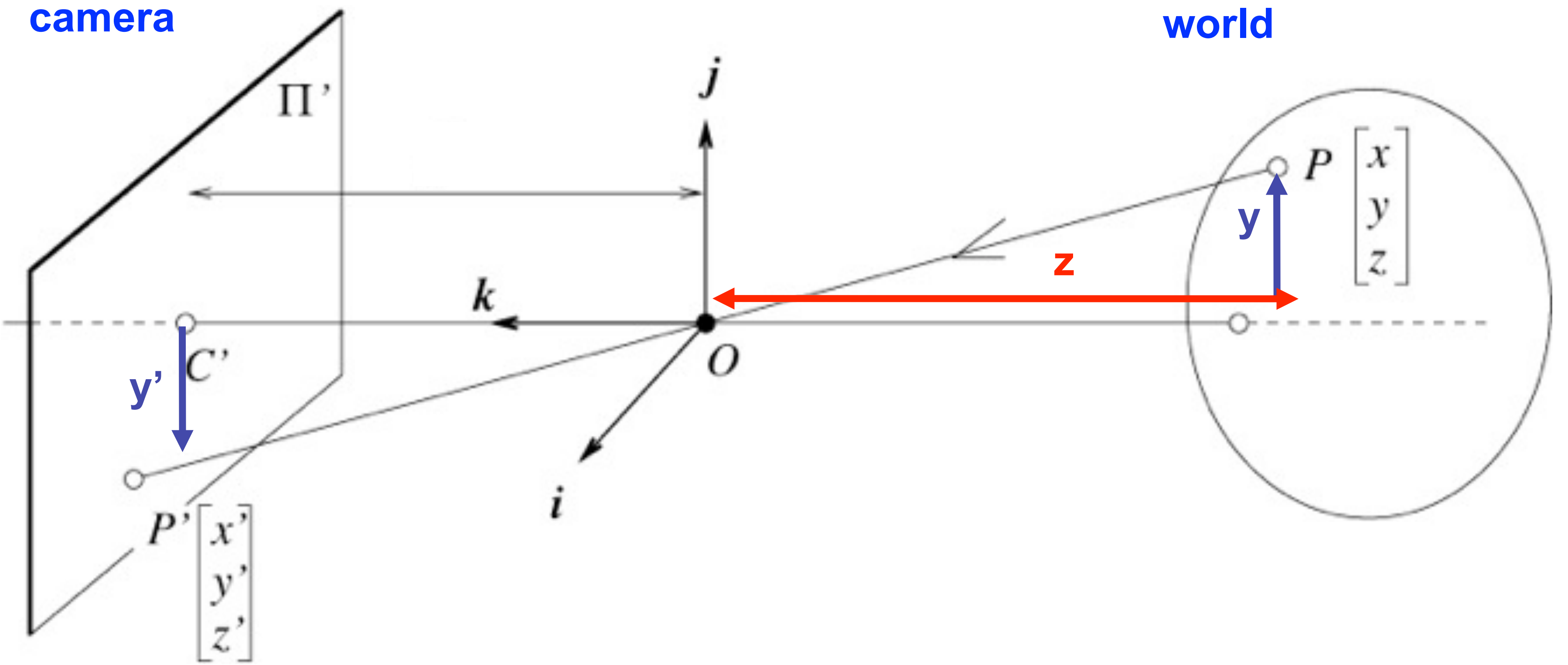
Perspective projection



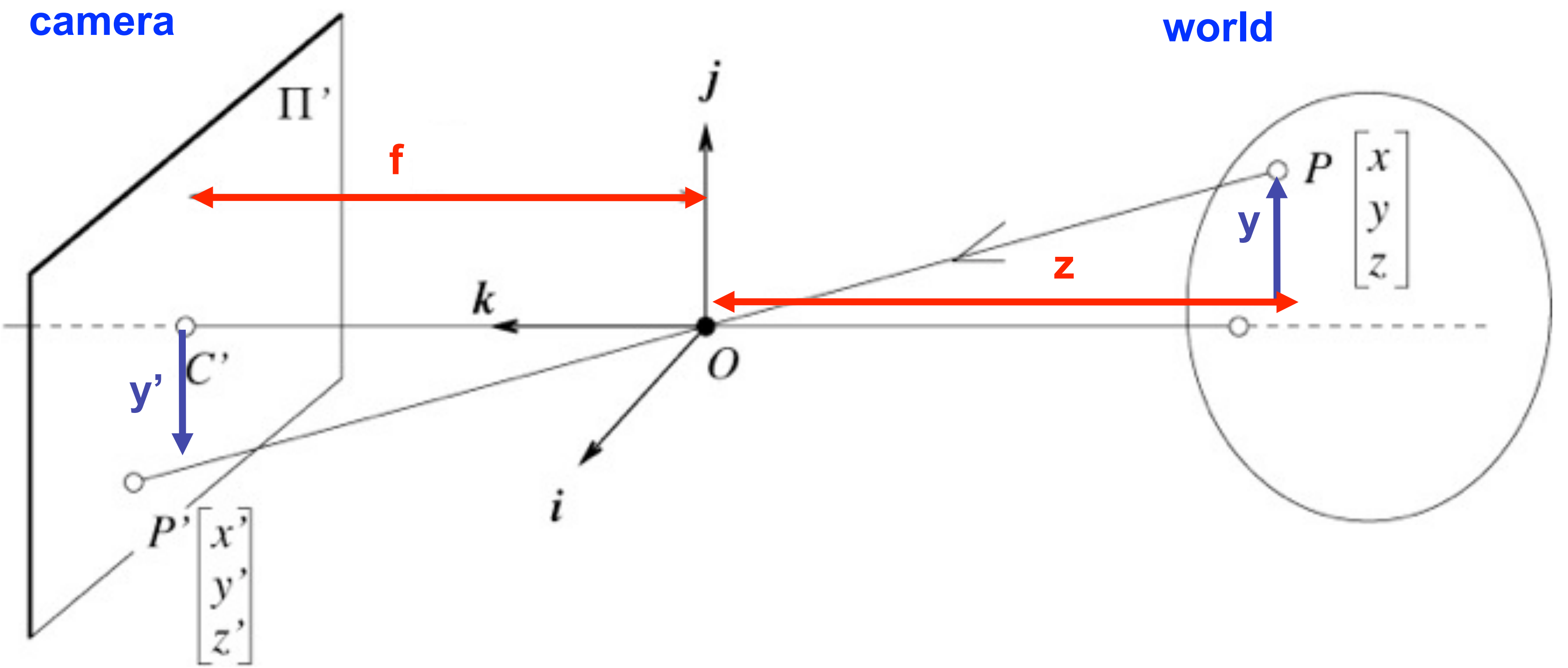
Perspective projection



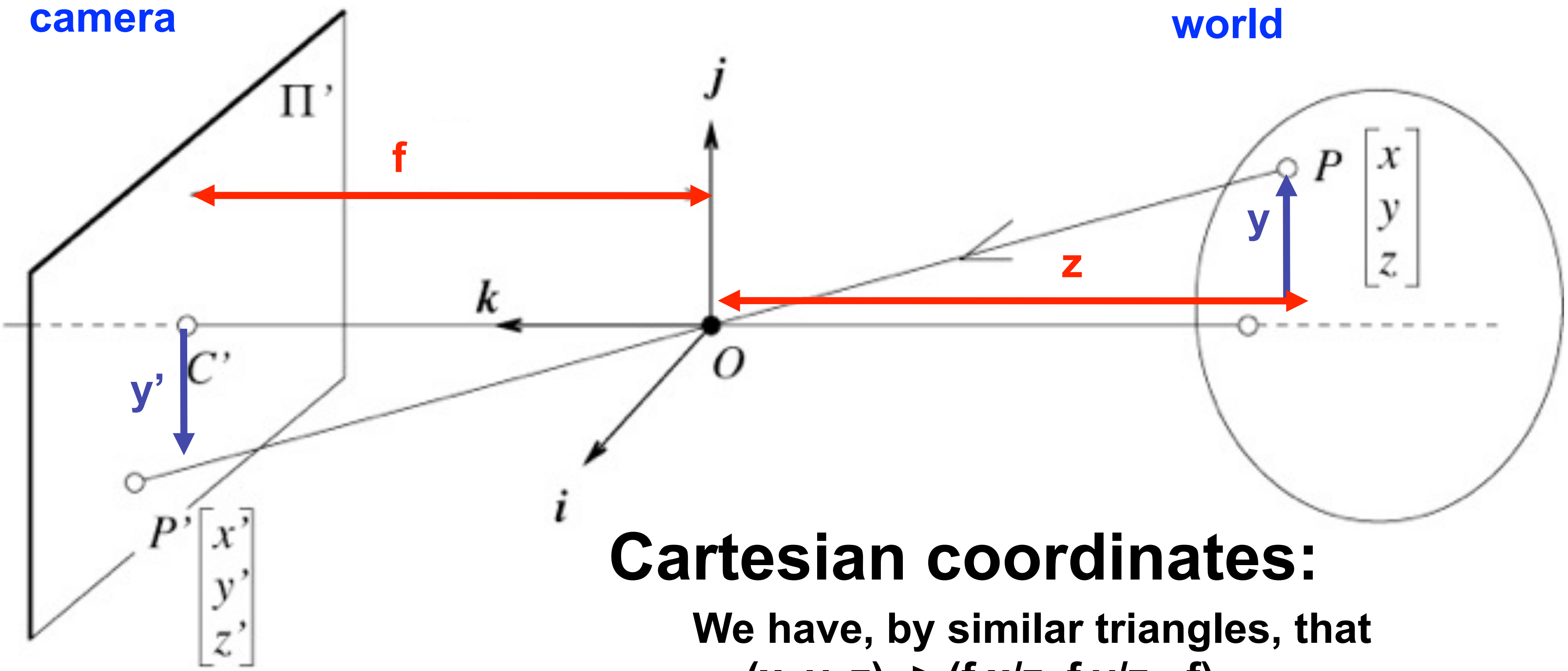
Perspective projection



Perspective projection



Perspective projection



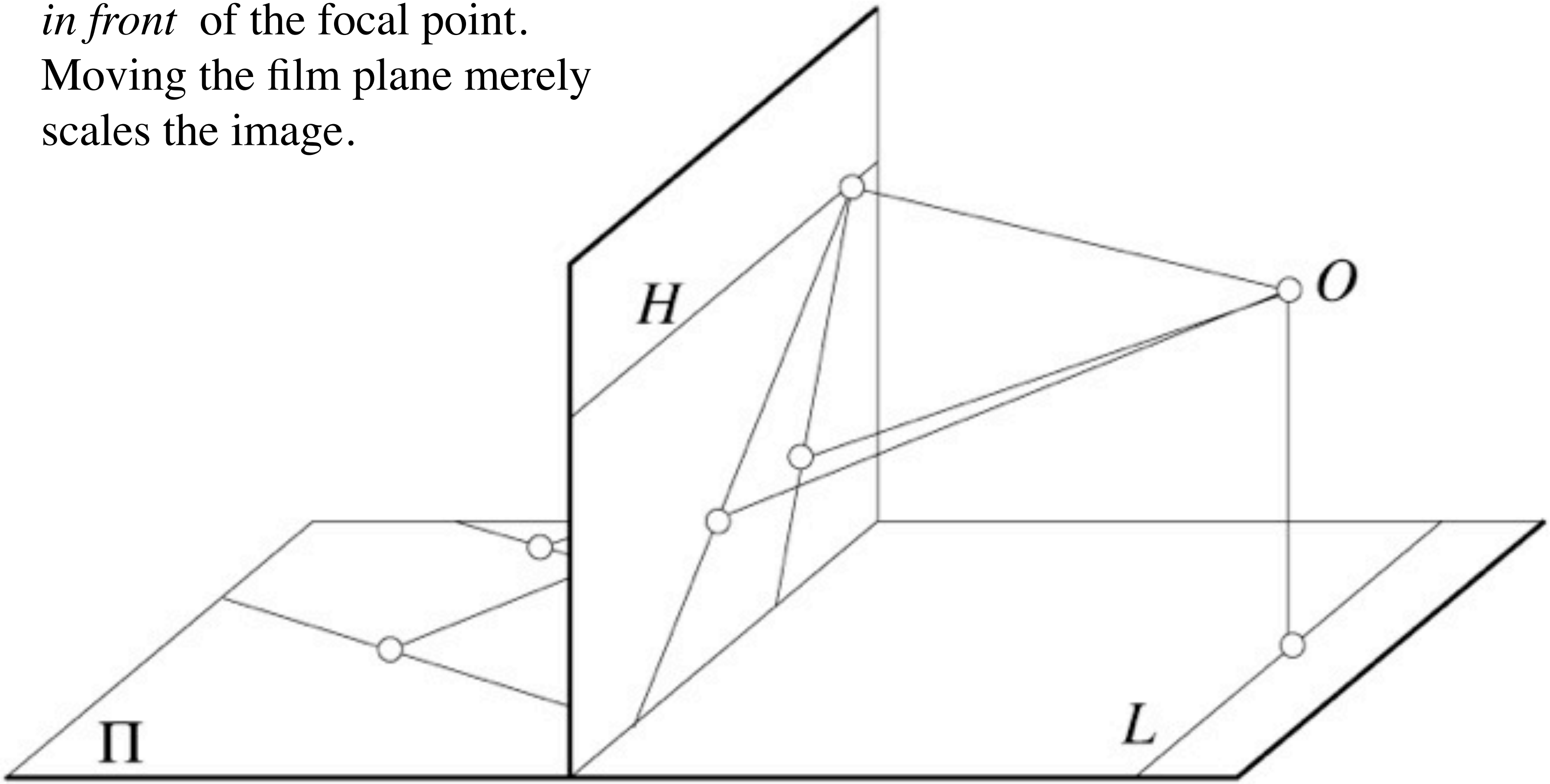
Cartesian coordinates:

We have, by similar triangles, that
(x, y, z) -> (f x/z, f y/z, -f)

Ignore the third coordinate, and get

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

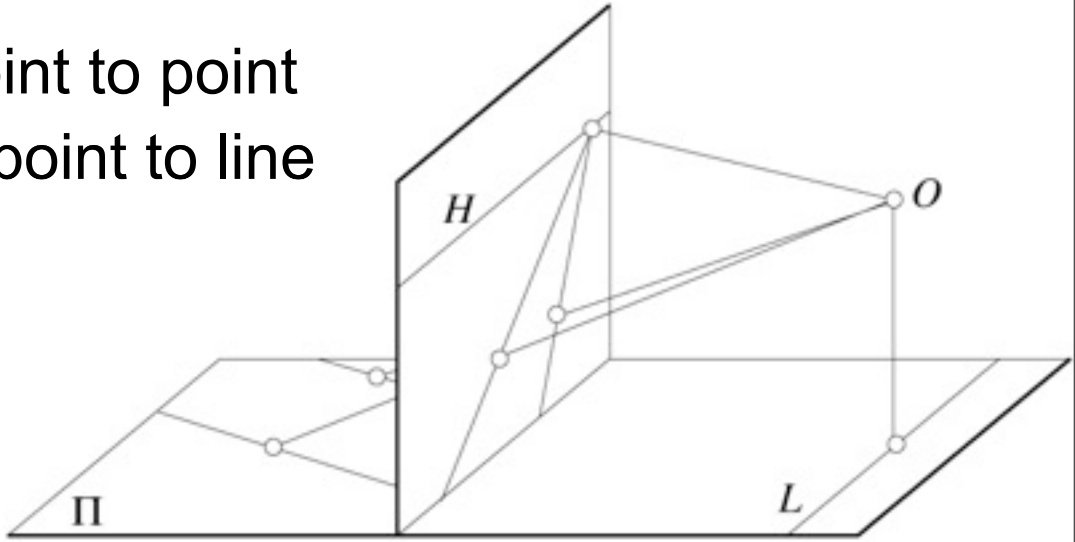
Common to draw film plane
in front of the focal point.
Moving the film plane merely
scales the image.



Forsyth&Ponce

Geometric properties of projection

- Points go to points
- Lines go to lines
- Planes go to whole image or half-planes.
- Polygons go to polygons
- Degenerate cases
  - line through focal point to point
  - plane through focal point to line



Line in 3-space

$$x(t) = x_0 + at$$

$$y(t) = y_0 + bt$$

$$z(t) = z_0 + ct$$

Line in 3-space

$$x(t) = x_0 + at$$

$$y(t) = y_0 + bt$$

$$z(t) = z_0 + ct$$

Perspective projection of that line

$$x'(t) = \frac{fx}{z} = \frac{f(x_0 + at)}{z_0 + ct}$$

$$y'(t) = \frac{fy}{z} = \frac{f(y_0 + bt)}{z_0 + ct}$$

Line in 3-space

$$x(t) = x_0 + at$$

$$y(t) = y_0 + bt$$

$$z(t) = z_0 + ct$$

Perspective projection of that line

$$x'(t) = \frac{fx}{z} = \frac{f(x_0 + at)}{z_0 + ct}$$

$$y'(t) = \frac{fy}{z} = \frac{f(y_0 + bt)}{z_0 + ct}$$

In the limit as $t \longrightarrow \pm\infty$
we have (for):

Line in 3-space

$$x(t) = x_0 + at$$

$$y(t) = y_0 + bt$$

$$z(t) = z_0 + ct$$

Perspective projection of that line

$$x'(t) = \frac{fx}{z} = \frac{f(x_0 + at)}{z_0 + ct}$$

$$y'(t) = \frac{fy}{z} = \frac{f(y_0 + bt)}{z_0 + ct}$$

In the limit as $t \longrightarrow \pm\infty$ **we have (for** $c \neq 0$ **):** $\longrightarrow$

$$x'(t) \longrightarrow \frac{fa}{c}$$

$$y'(t) \longrightarrow \frac{fb}{c}$$

Line in 3-space

$$x(t) = x_0 + at$$

$$y(t) = y_0 + bt$$

$$z(t) = z_0 + ct$$

Perspective projection of that line

$$x'(t) = \frac{fx}{z} = \frac{f(x_0 + at)}{z_0 + ct}$$

$$y'(t) = \frac{fy}{z} = \frac{f(y_0 + bt)}{z_0 + ct}$$

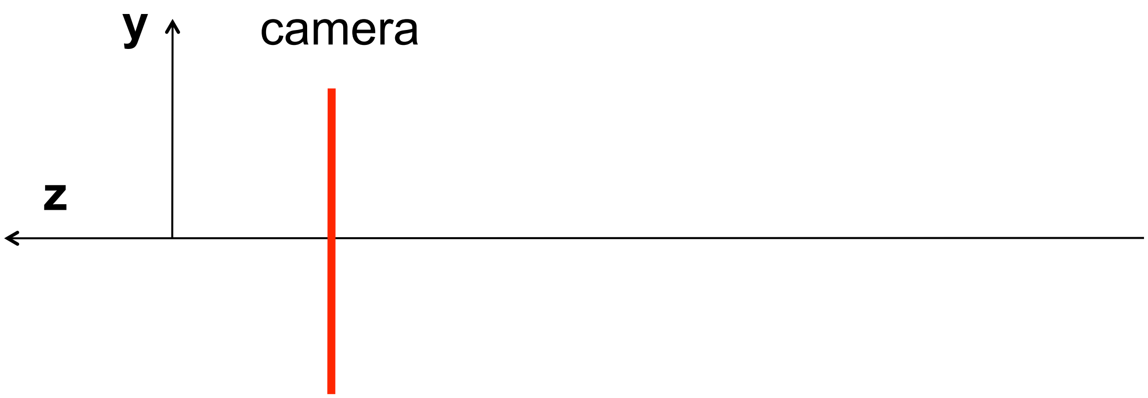
In the limit as $t \longrightarrow \pm\infty$ we have (for $c \neq 0$):

$$x'(t) \longrightarrow \frac{fa}{c}$$

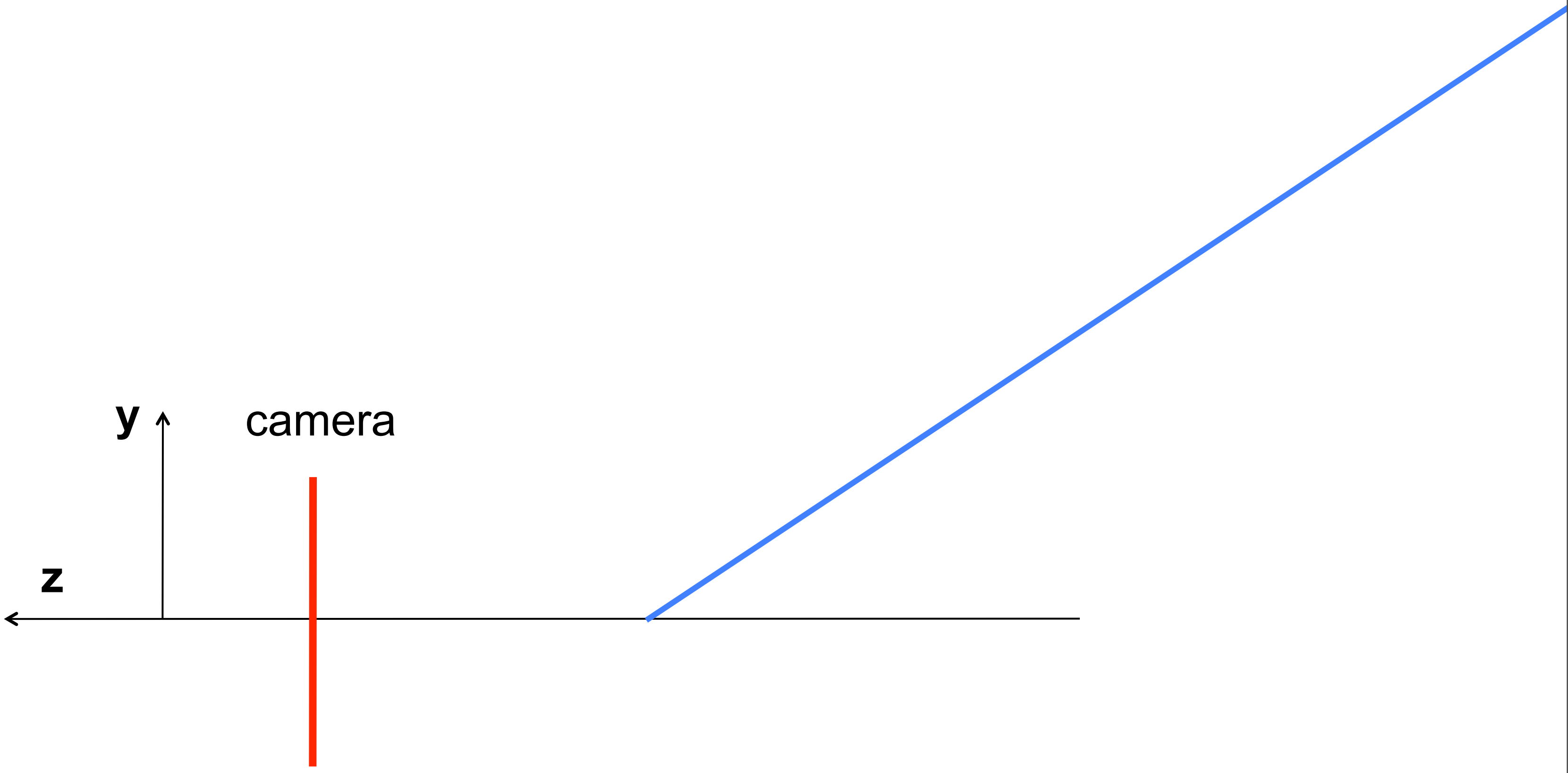
$$y'(t) \longrightarrow \frac{fb}{c}$$

This tells us that any set of parallel lines (same a, b, c parameters) project to the same point (called the vanishing point).

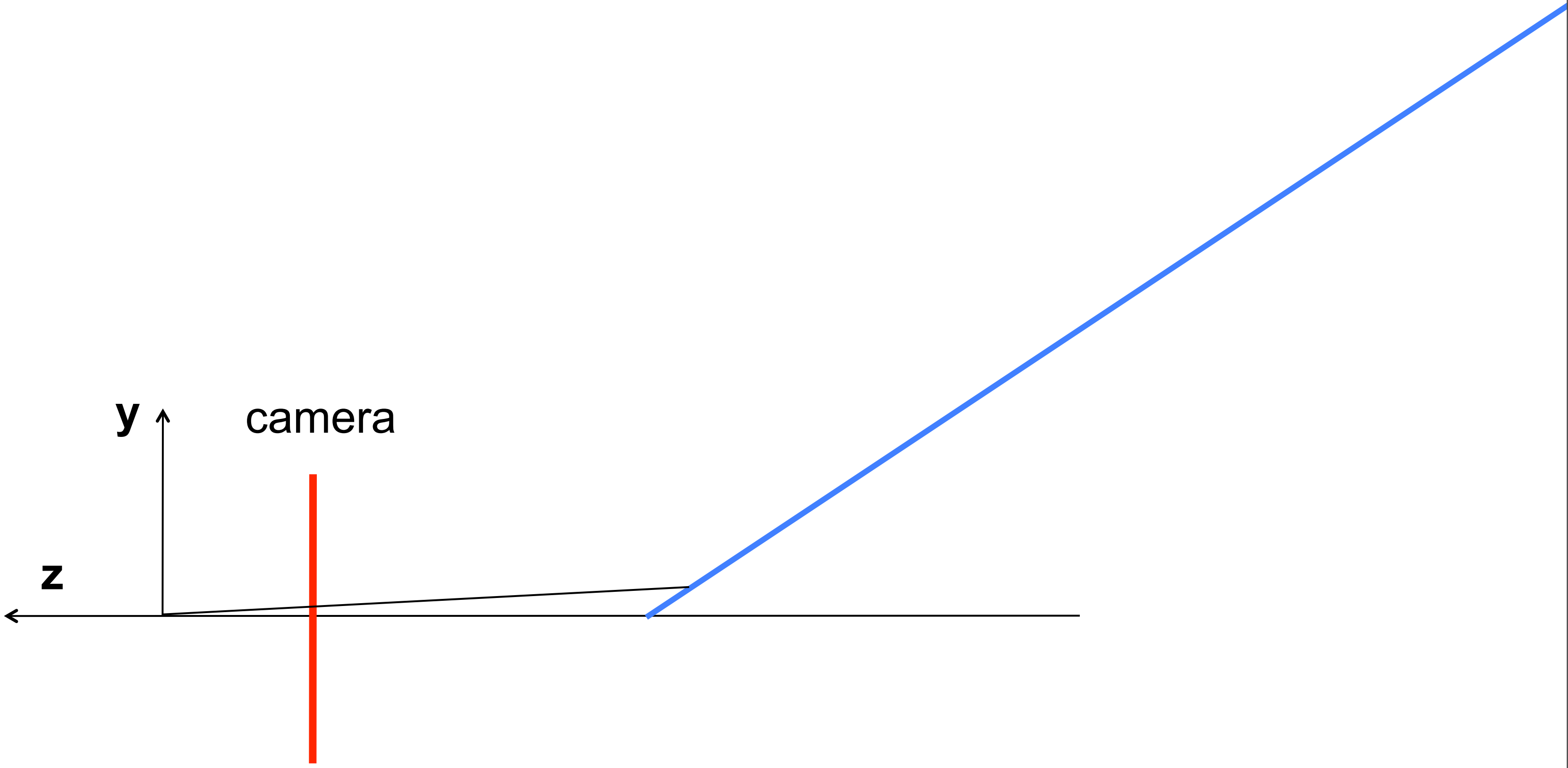
Vanishing point



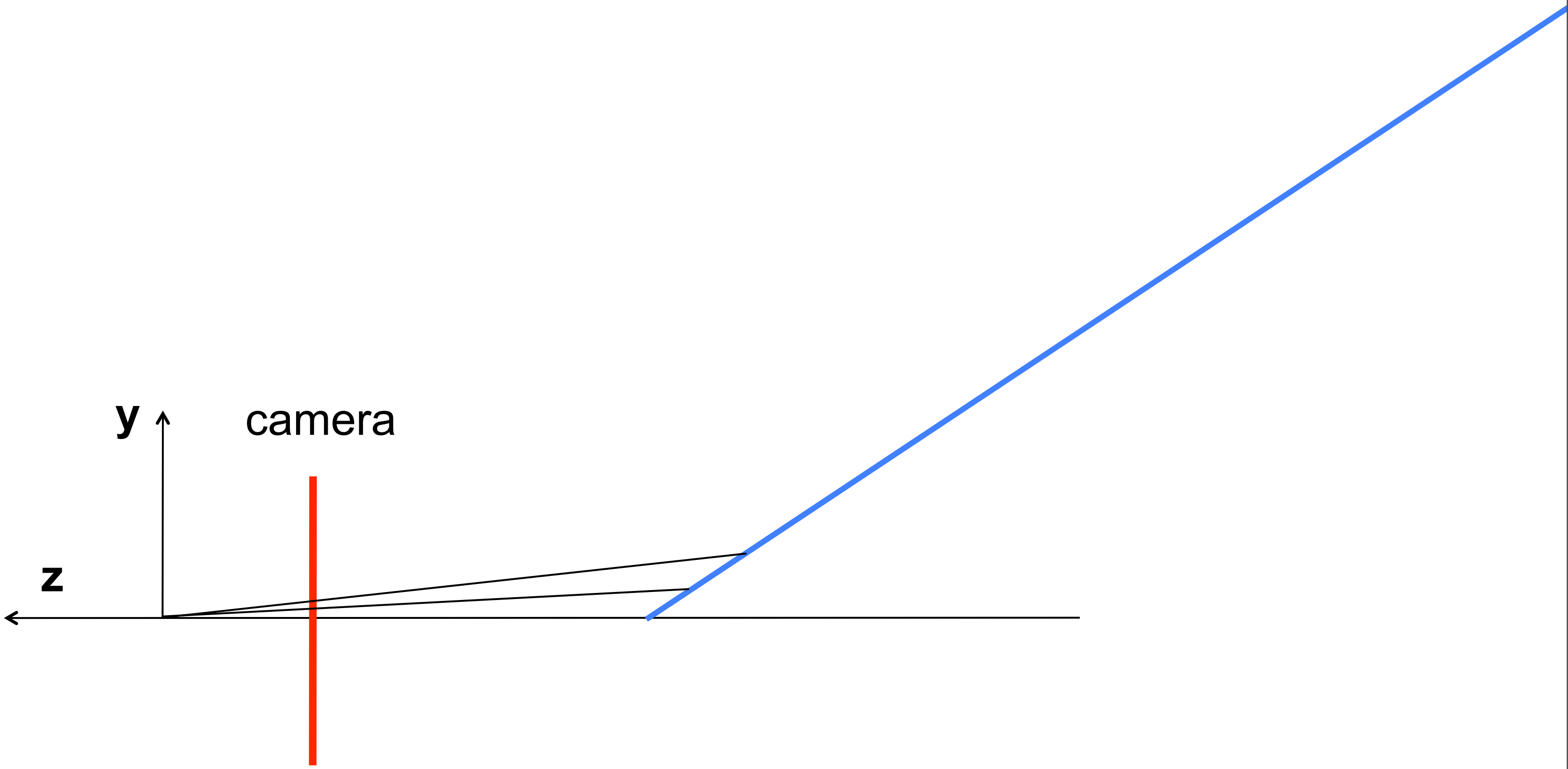
Vanishing point



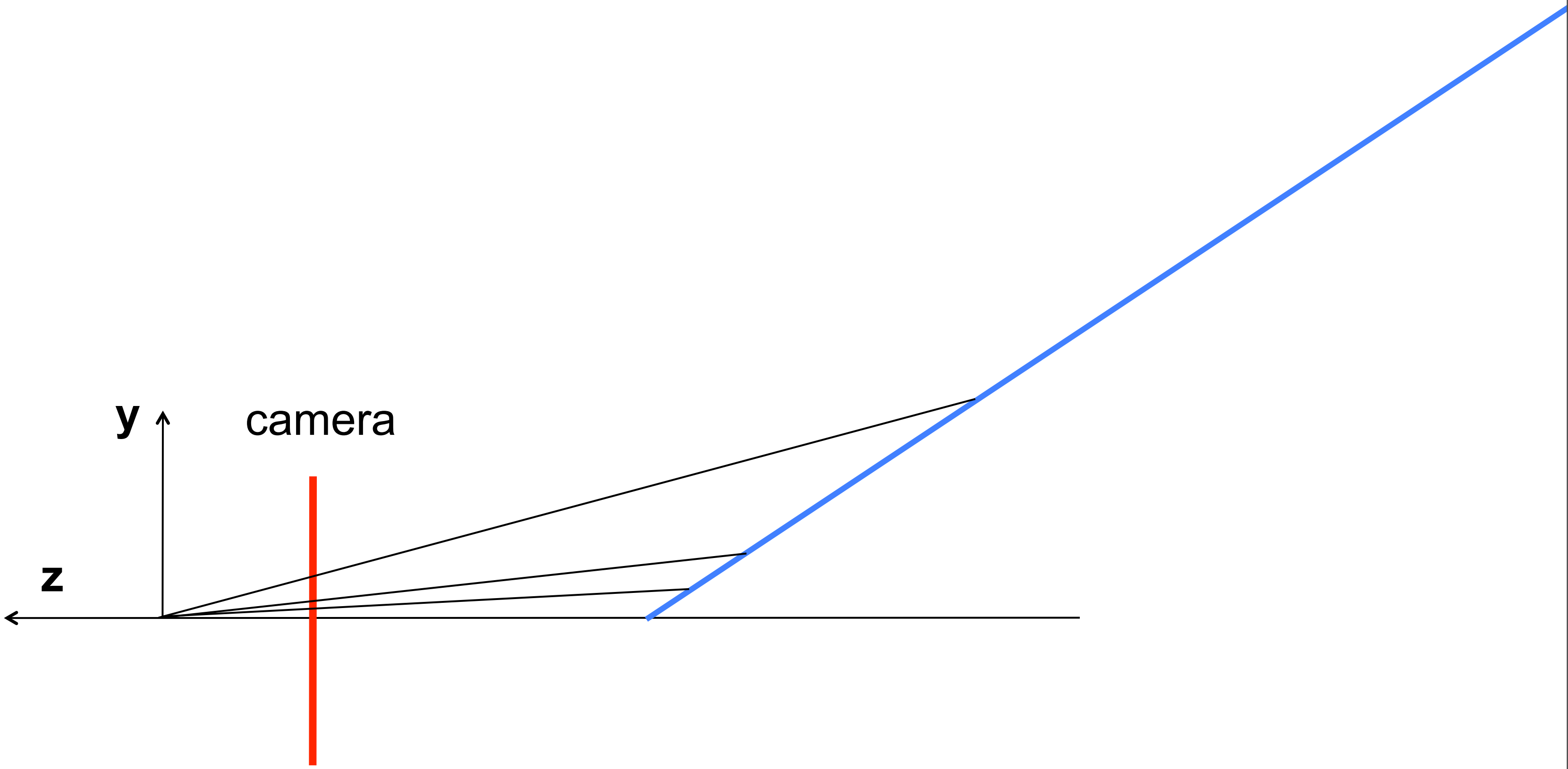
Vanishing point



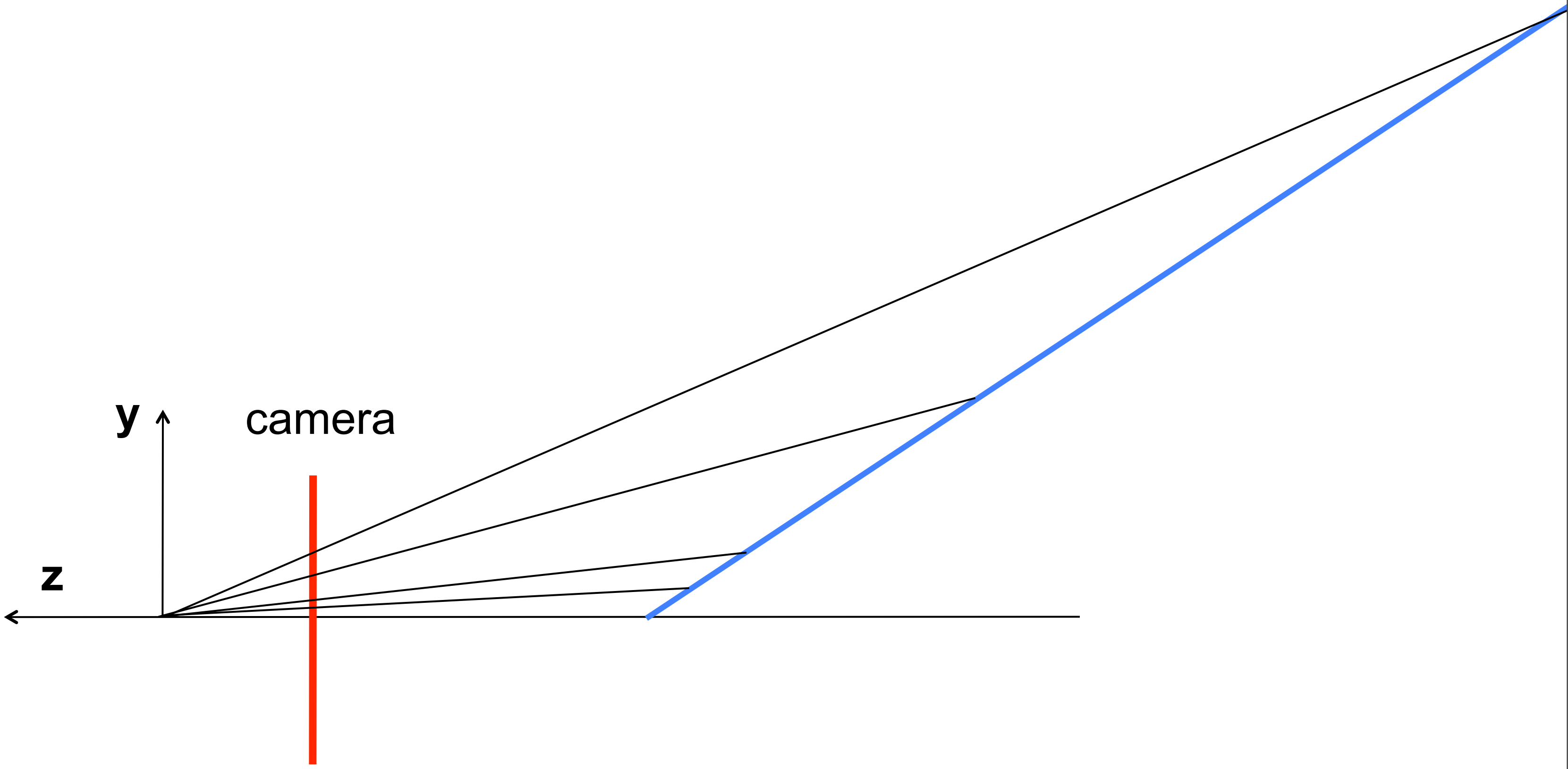
Vanishing point



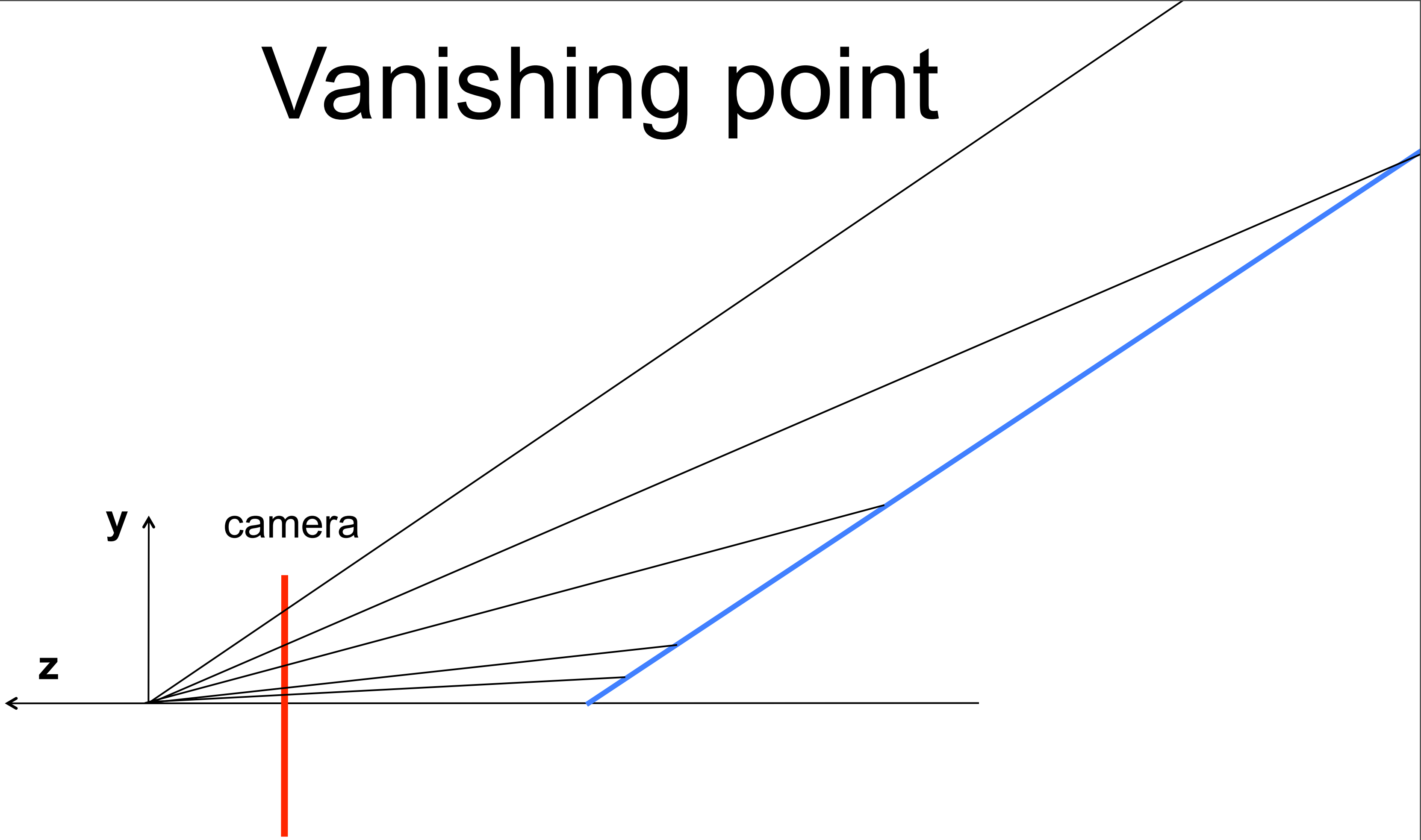
Vanishing point



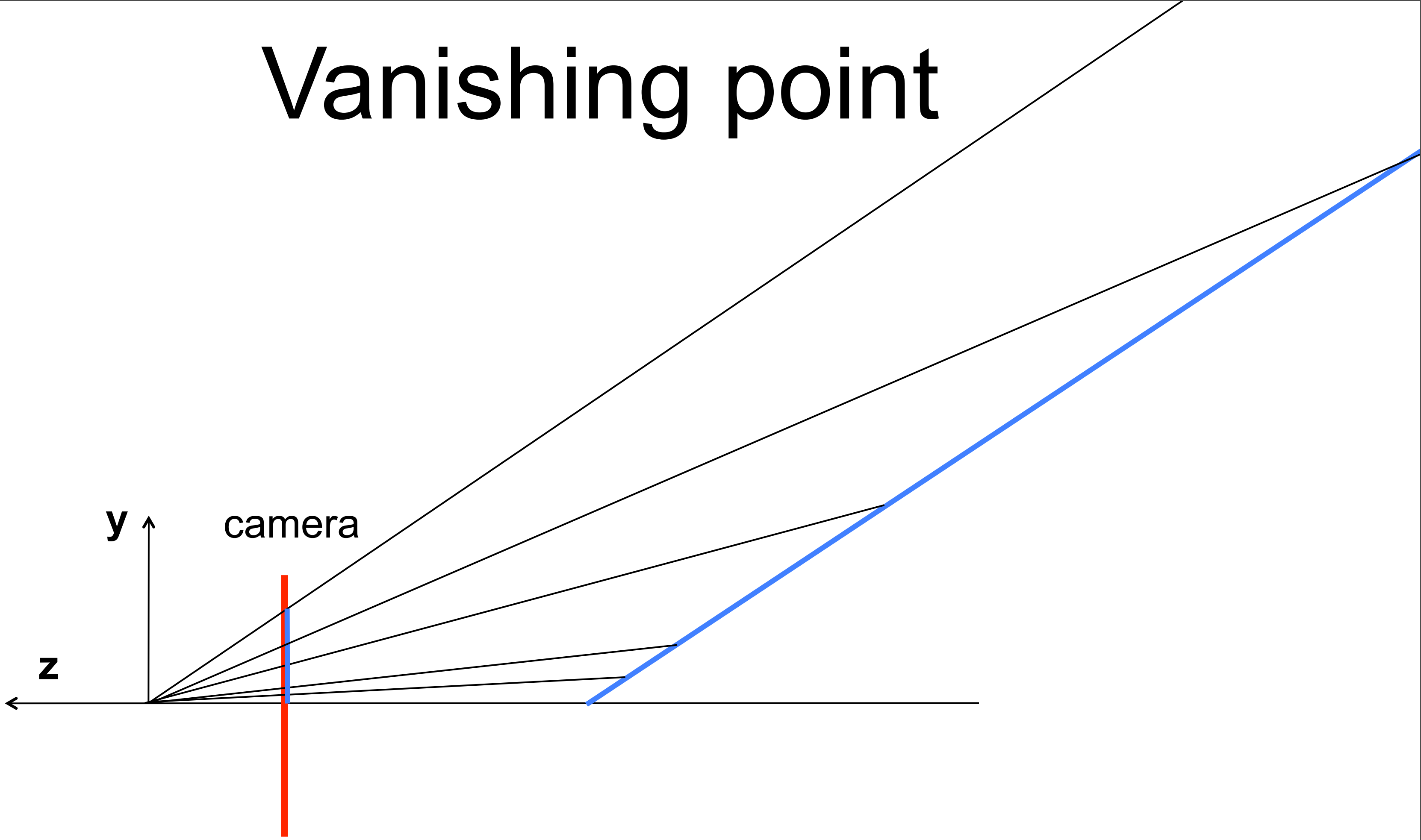
Vanishing point

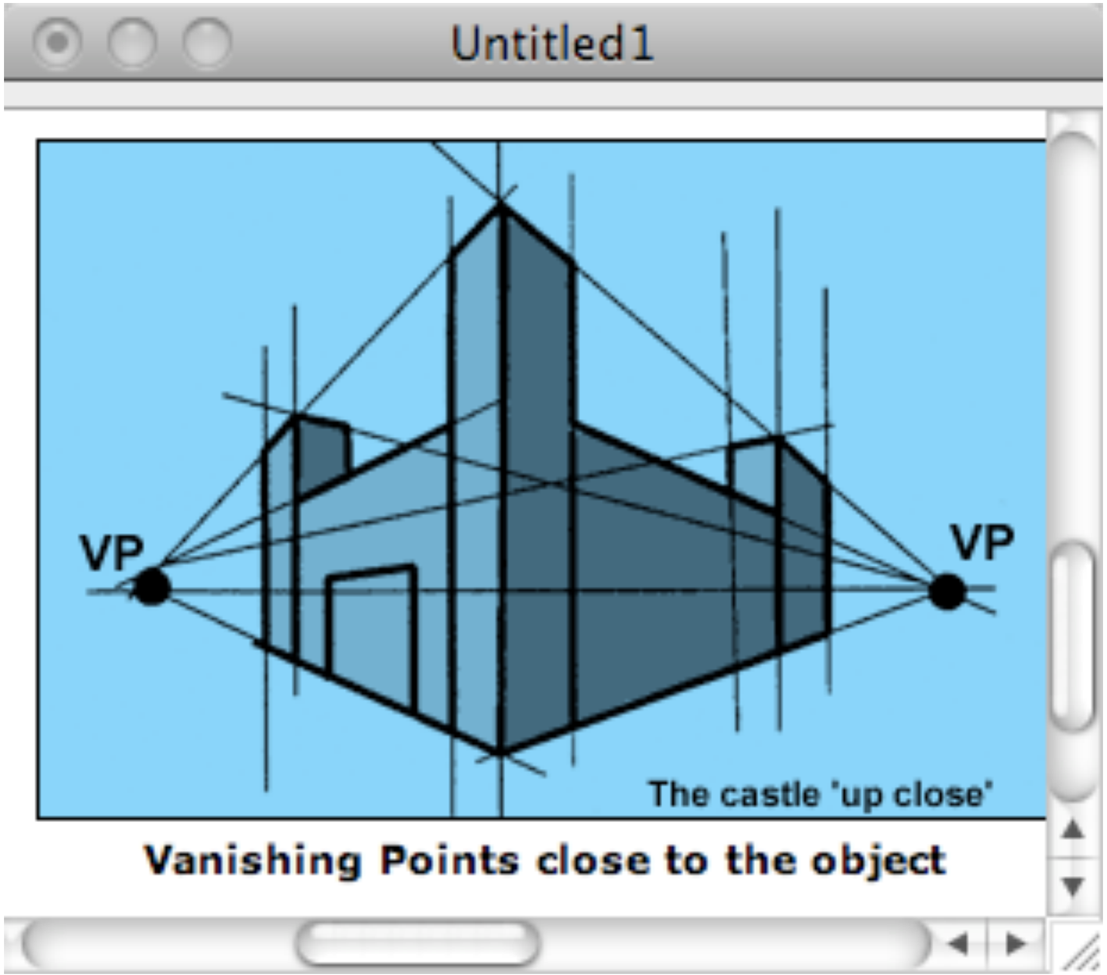


Vanishing point



Vanishing point



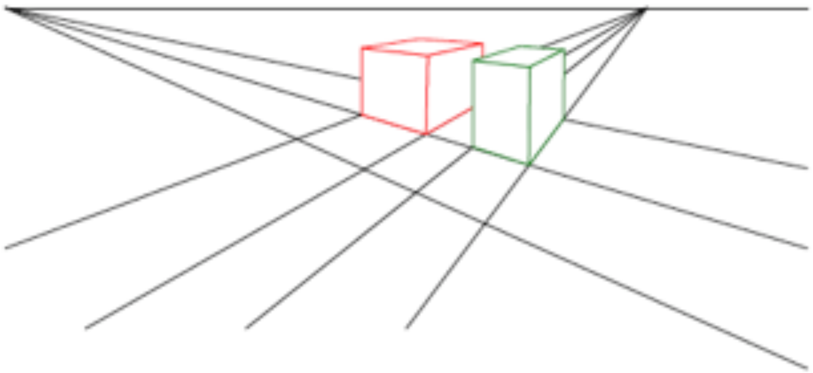


Vanishing Points close to the object

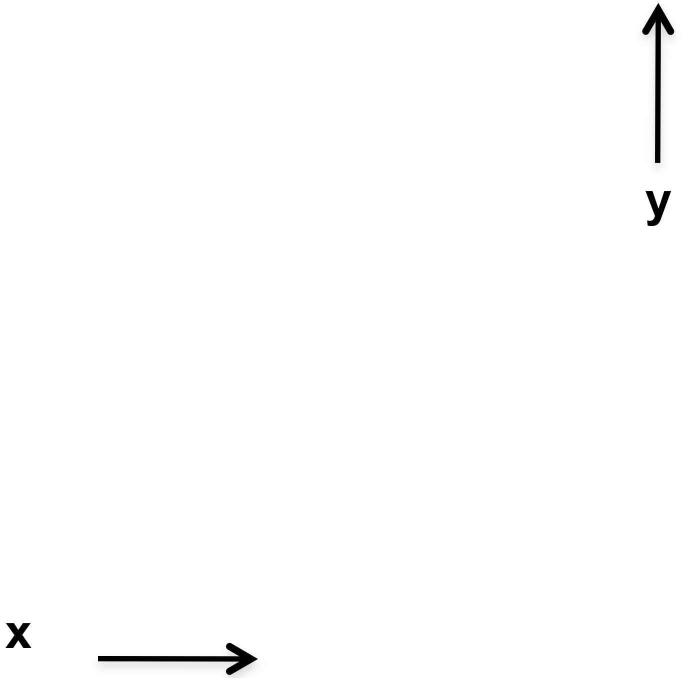
http://www.ider.herts.ac.uk/school/courseware/graphics/two_point_perspective.html

Vanishing points

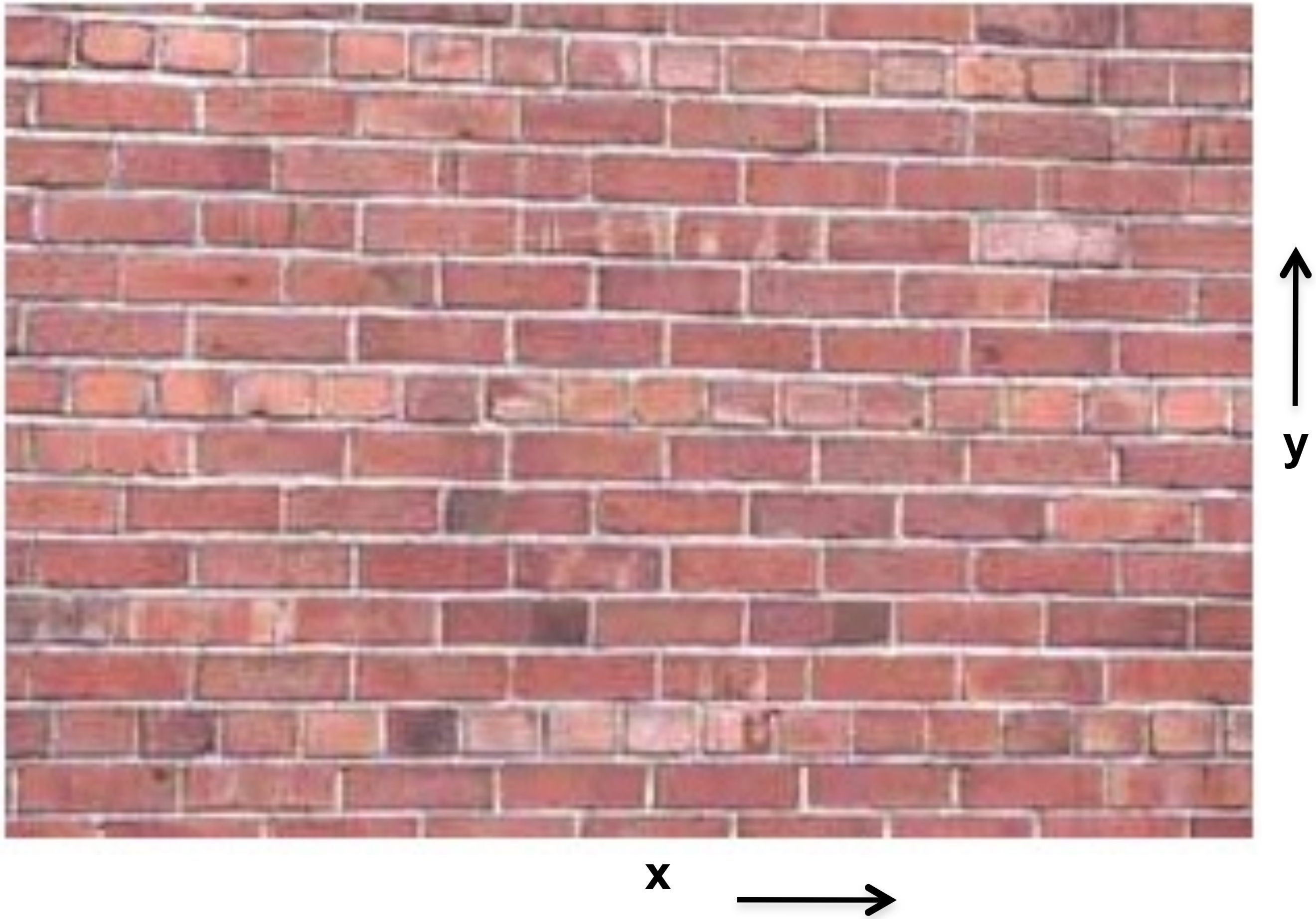
- Each set of parallel lines (=direction) meets at a different point
  - The *vanishing point* for this direction
- Sets of parallel lines on the same plane lead to *collinear* vanishing points.
  - The line is called the *horizon* for that plane



What if you photograph a brick wall head-on?



What if you photograph a brick wall head-on?



Brick wall line in 3-space

$$x(t) = x_0 + at$$

$$y(t) = y_0$$

$$z(t) = z_0$$

Brick wall line in 3-space

$$x(t) = x_0 + at$$

$$y(t) = y_0$$

$$z(t) = z_0$$

Perspective projection of that line

$$x'(t) = \frac{f \cdot (x_0 + at)}{z_0}$$

$$y'(t) = \frac{f \cdot y_0}{z_0}$$

Brick wall line in 3-space

$$x(t) = x_0 + at$$

$$y(t) = y_0$$

$$z(t) = z_0$$

Perspective projection of that line

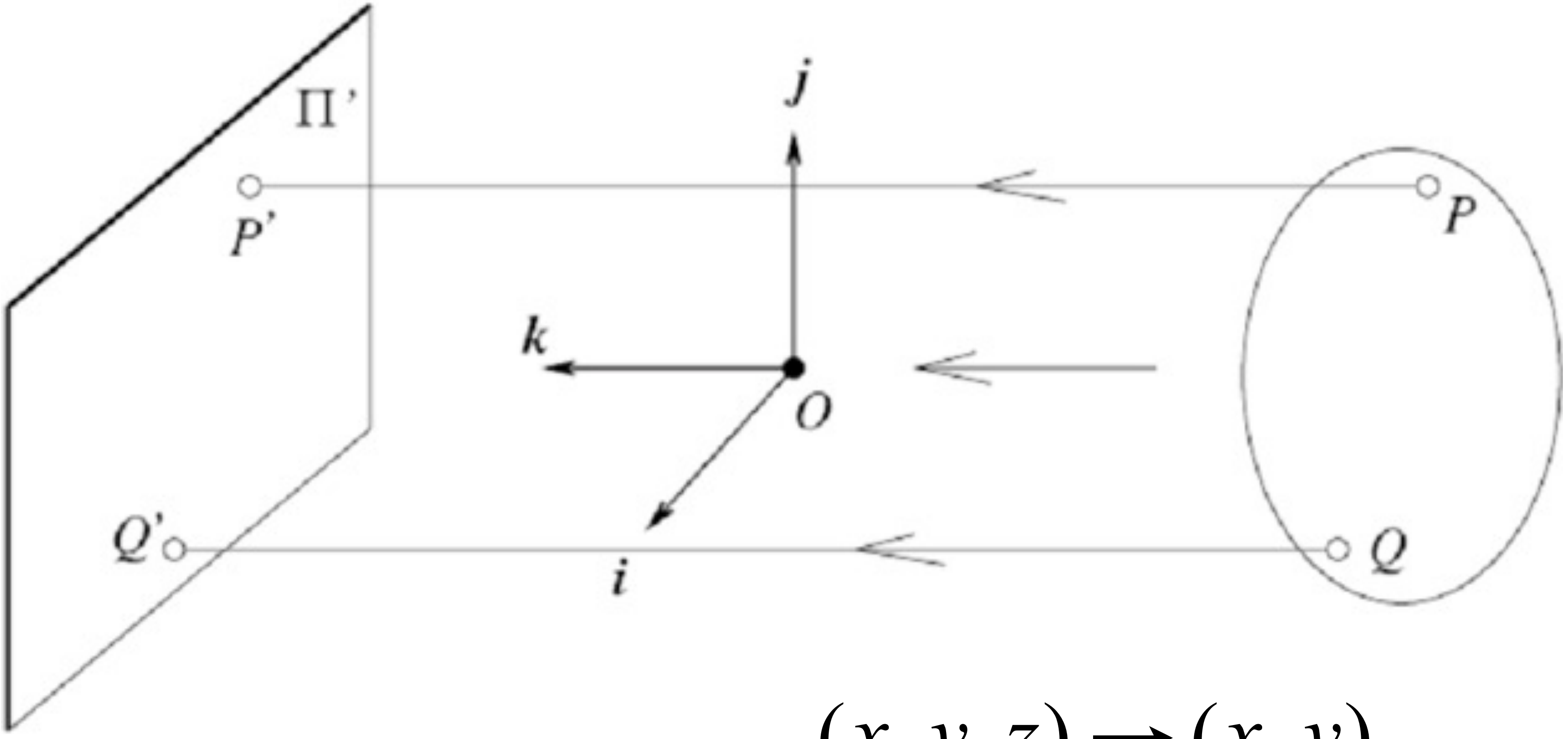
$$x'(t) = \frac{f \cdot (x_0 + at)}{z_0}$$

$$y'(t) = \frac{f \cdot y_0}{z_0}$$

All bricks have same z_0. Those in same row have same y_0

Thus, a brick wall, photographed head-on, gets rendered as set of parallel lines in the image plane.

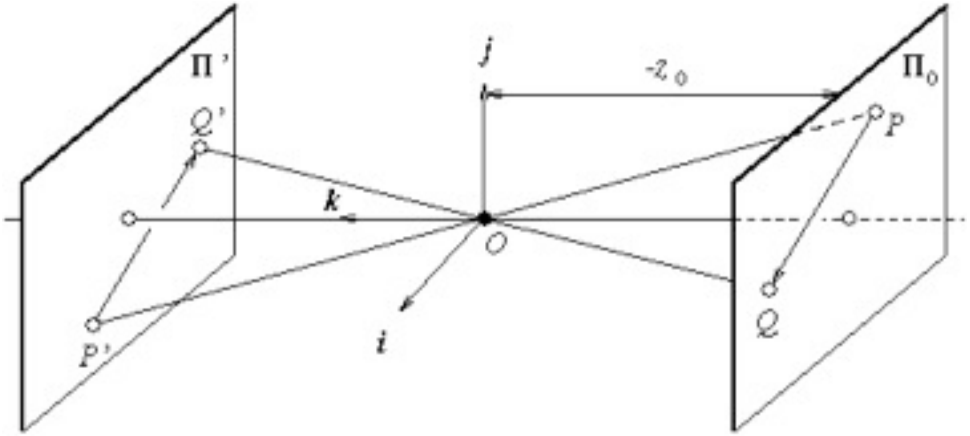
Other projection models: Orthographic projection



$$(x, y, z) \rightarrow (x, y)$$

Other projection models: Weak perspective

- Issue
  - perspective effects, but not over the scale of individual objects
  - collect points into a group at about the same depth, then divide each point by the depth of its group
  - Adv: easy
  - Disadv: only approximate



$$(x, y, z) \rightarrow \left(\frac{fx}{z_0}, \frac{fy}{z_0} \right)$$

Three camera projections

3-d point **2-d image position**

(1) Perspective: $$(x, y, z) \rightarrow \left(\frac{fx}{z}, \frac{fy}{z} \right)$$

(2) Weak perspective: $$(x, y, z) \rightarrow \left(\frac{fx}{z_0}, \frac{fy}{z_0} \right)$$

(3) Orthographic: $$(x, y, z) \rightarrow (x, y)$$

Homogeneous coordinates

Is the perspective projection a linear transformation?

Slide by Steve Seitz

Homogeneous coordinates

Is the perspective projection a linear transformation?

no—division by z is nonlinear

Slide by Steve Seitz

Homogeneous coordinates

Is the perspective projection a linear transformation?

no—division by z is nonlinear

Trick: add one more coordinate:

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

homogeneous image coordinates **homogeneous scene coordinates**

Converting *from* homogeneous coordinates

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

Slide by Steve Seitz

Perspective Projection

- Projection is a matrix multiply using homogeneous coordinates:

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

This is known as perspective projection

- **The matrix is the projection matrix**

Slide by Steve Seitz

Perspective Projection

How does scaling the projection matrix change the transformation?

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

Slide by Steve Seitz

Perspective Projection

How does scaling the projection matrix change the transformation?

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

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

Slide by Steve Seitz

Perspective Projection

How does scaling the projection matrix change the transformation?

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

$$\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} fx \\ fy \\ z \end{bmatrix}$$

Slide by Steve Seitz

Perspective Projection

How does scaling the projection matrix change the transformation?

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

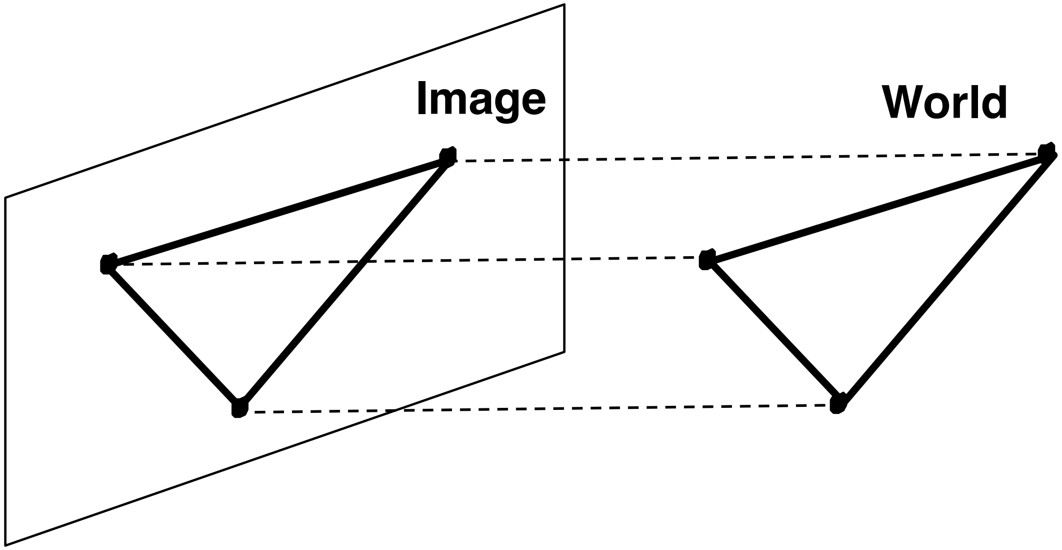
$$\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} fx \\ fy \\ z \end{bmatrix} \Rightarrow \left(f\frac{x}{z}, f\frac{y}{z} \right)$$

Slide by Steve Seitz

Orthographic Projection

Special case of perspective projection

- Distance from the COP to the PP is infinite



- Also called “parallel projection”
- What’s the projection matrix?

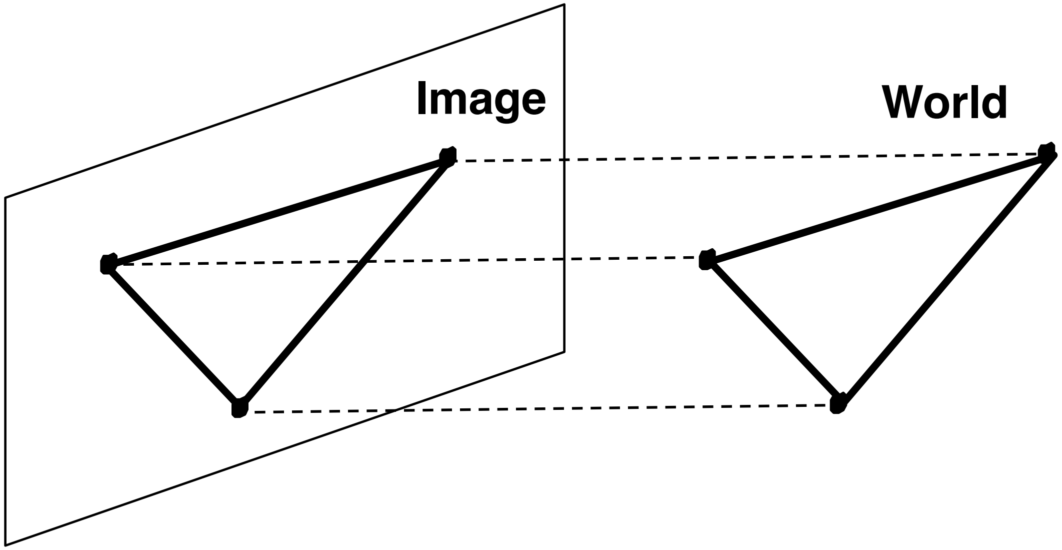
?

Slide by Steve Seitz

Orthographic Projection

Special case of perspective projection

- Distance from the COP to the PP is infinite



- Also called “parallel projection”
- What’s the projection matrix?

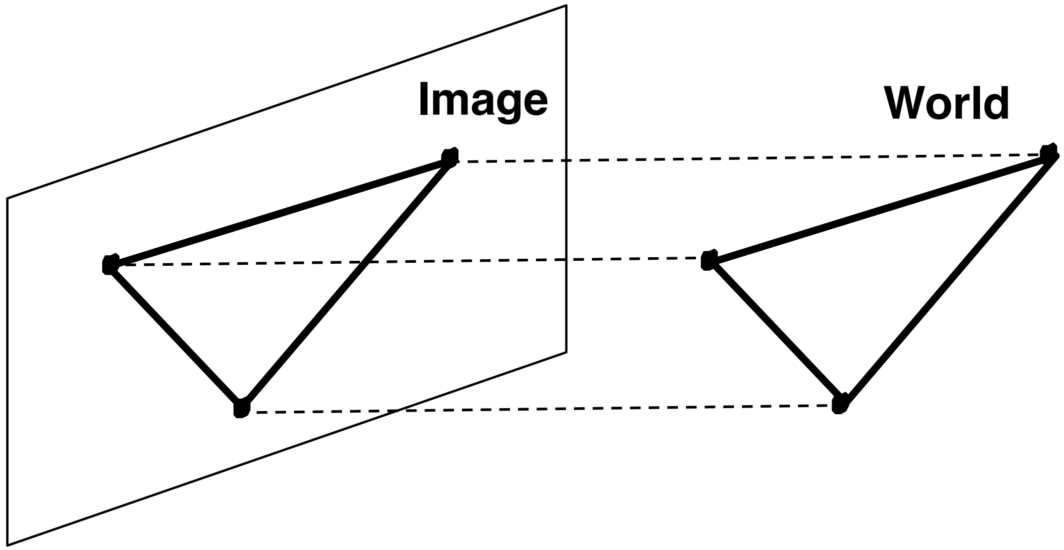
$$\boxed{\quad ? \quad} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \Rightarrow (x, y)$$

Slide by Steve Seitz

Orthographic Projection

Special case of perspective projection

- Distance from the COP to the PP is infinite



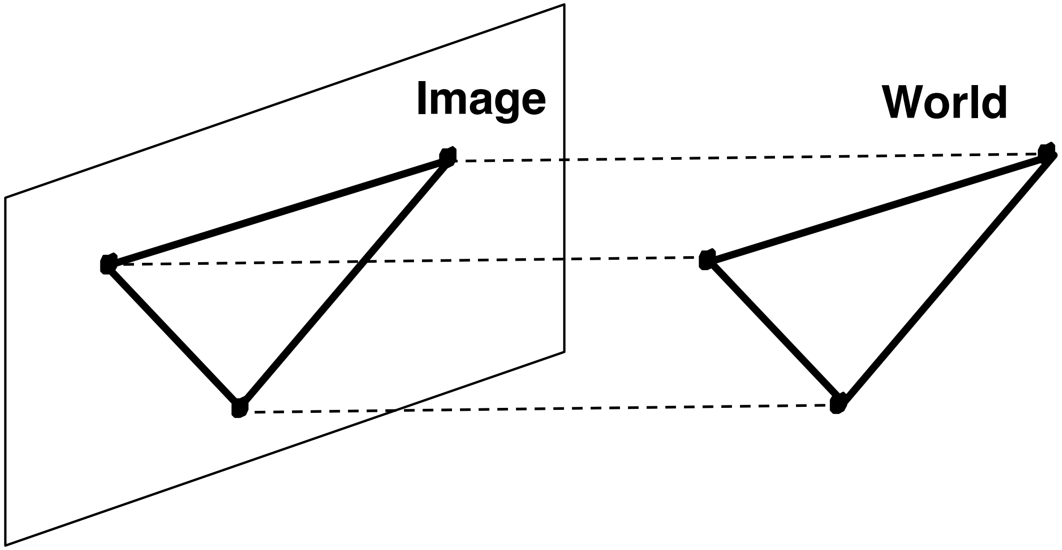
- Also called “parallel projection”
- What’s the projection matrix?

Slide by Steve Seitz

Orthographic Projection

Special case of perspective projection

- Distance from the COP to the PP is infinite



- Also called "parallel projection"
- What's the projection matrix?

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \Rightarrow (x, y)$$

Slide by Steve Seitz

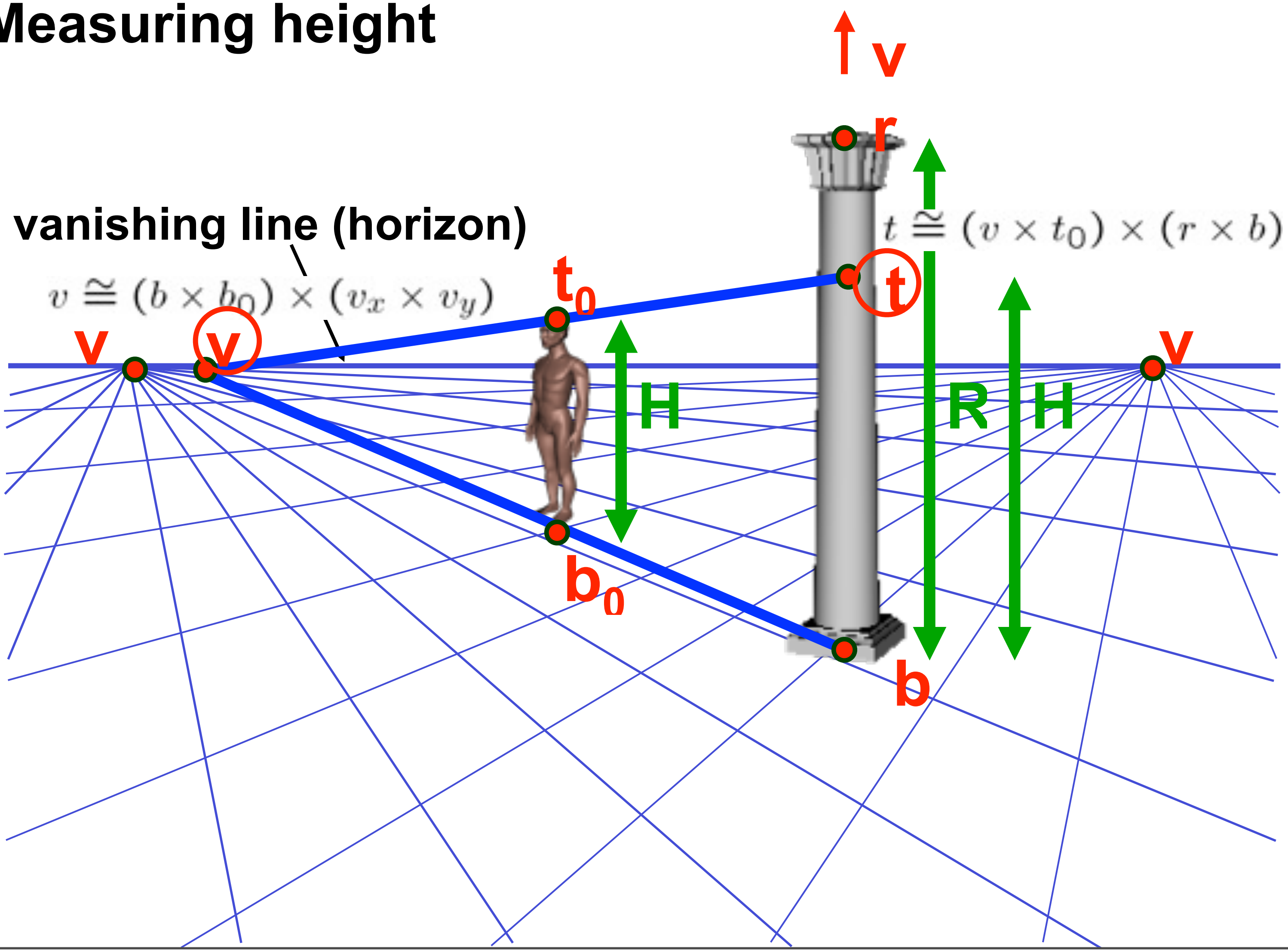
Camera calibration

Use the camera to tell you things about the world:

- Relationship between coordinates in the world and coordinates in the image: *geometric camera calibration, see* Szeliski, section 5.2, 5.3 for references
- (Relationship between intensities in the world and intensities in the image: *photometric image formation*, see Szeliski, sect. 2.2.)

One reason to calibrate a camera

Measuring height

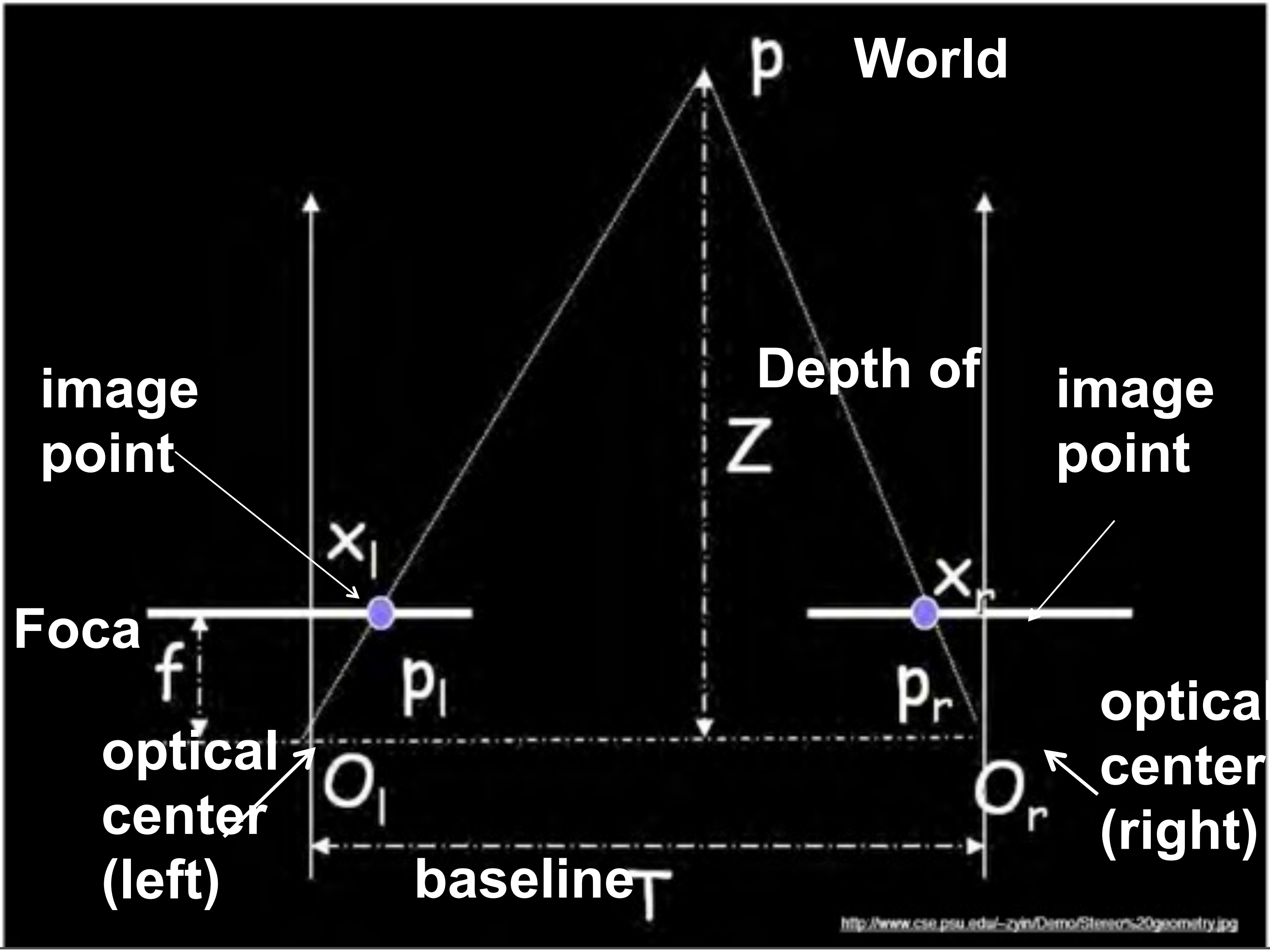


vanishing line (horizon)

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

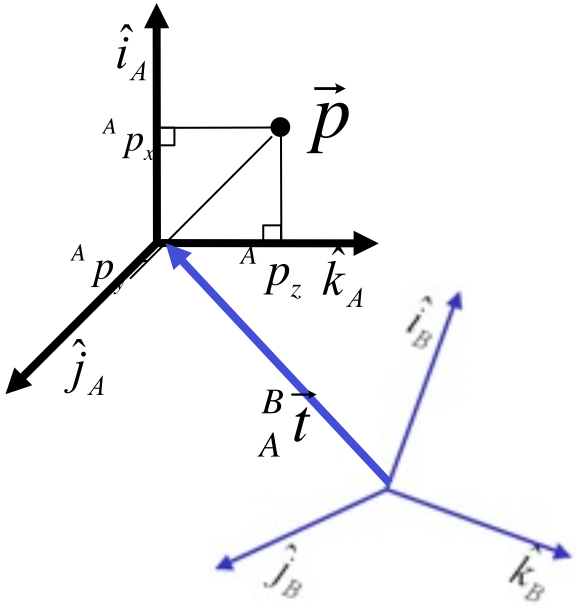
$$t \cong (v \times t_0) \times (r \times b)$$

Another reason to calibrate a camera



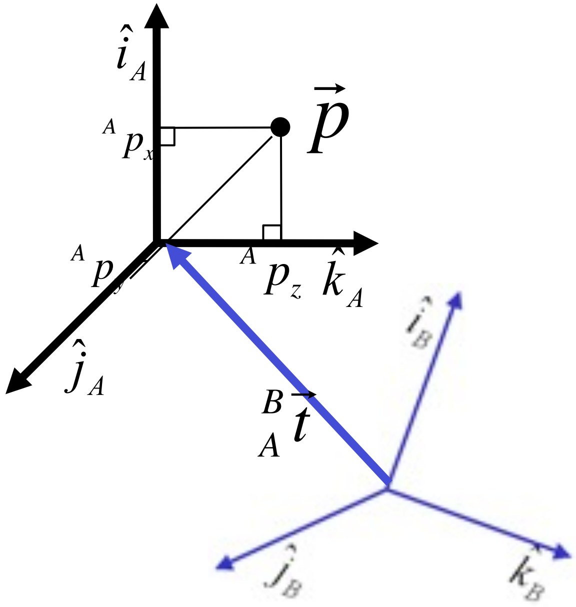
http://www.cse.psu.edu/~zyin/Demo/Stereo%20geometry.jpg

Translation and rotation



Translation and rotation

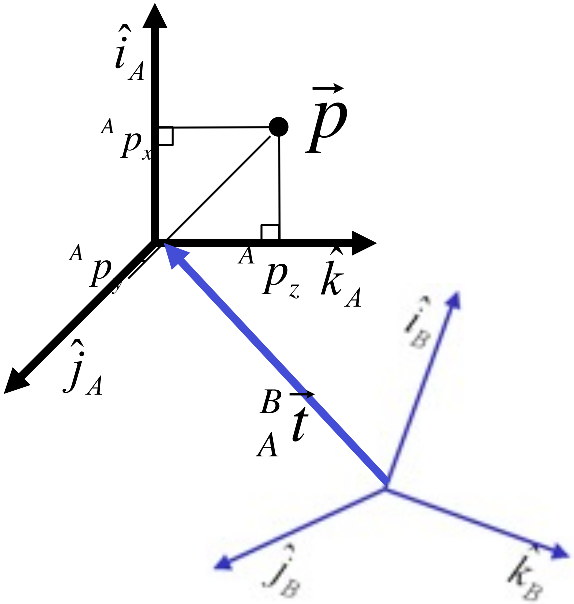
$$ {}^{B}\vec{p} = {}^{B}_{A}R \; {}^{A}\vec{p} + {}^{B}_{A}\vec{t} $$



Translation and rotation

“as described in the coordinates of frame B”

$$ {}^{B}\vec{p} = {}^{B}_{A}R\ {}^{A}\vec{p} + {}^{B}_{A}\vec{t} $$



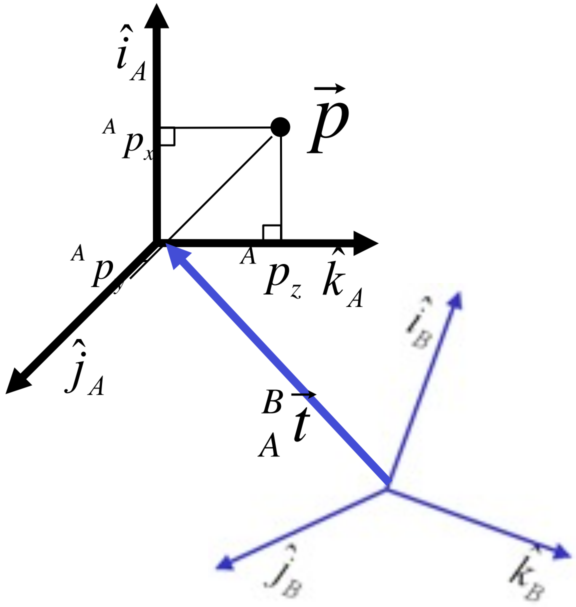
Translation and rotation

“as described in the coordinates of frame B”

Let’s write

$${}^{B}\vec{p} = {}^{B}_{A}R\;{}^{A}\vec{p} + {}^{B}_{A}\vec{t}$$

as a single matrix equation:

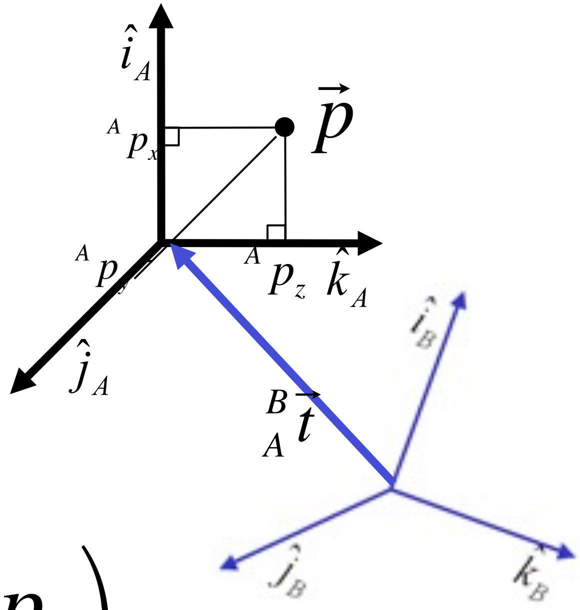


Translation and rotation

“as described in the coordinates of frame B”

Let’s write

$$ {}^{B}\vec{p} = {}^{B}_{A}R\,{}^{A}\vec{p} + {}^{B}_{A}\vec{t} $$



as a single matrix equation:

$$\begin{pmatrix} {}^{B}p_x \\ {}^{B}p_y \\ {}^{B}p_z \\ 1 \end{pmatrix} = \begin{pmatrix} \boxed{\begin{matrix} - & - & - \\ - & {}^{B}_{A}R & - \\ - & - & - \end{matrix}} & \boxed{\begin{matrix} | \\ {}^{B}_{A}t \\ | \end{matrix}} \\ \begin{matrix} 0 & \quad 0 & \quad 0 \end{matrix} & 1 \end{pmatrix} \begin{pmatrix} {}^{A}p_x \\ {}^{A}p_y \\ {}^{A}p_z \\ 1 \end{pmatrix}$$

Translation and rotation, written in each set of coordinates

Non-homogeneous coordinates

$$ {}^{B}\vec{p} = {}^{B}_{A}R \; {}^{A}\vec{p} + {}^{B}_{A}\vec{t} $$

Translation and rotation, written in each set of coordinates

Non-homogeneous coordinates

$$ {}^{B}\vec{p} = {}^{B}_{A}R\;{}^{A}\vec{p} + {}^{B}_{A}\vec{t} $$

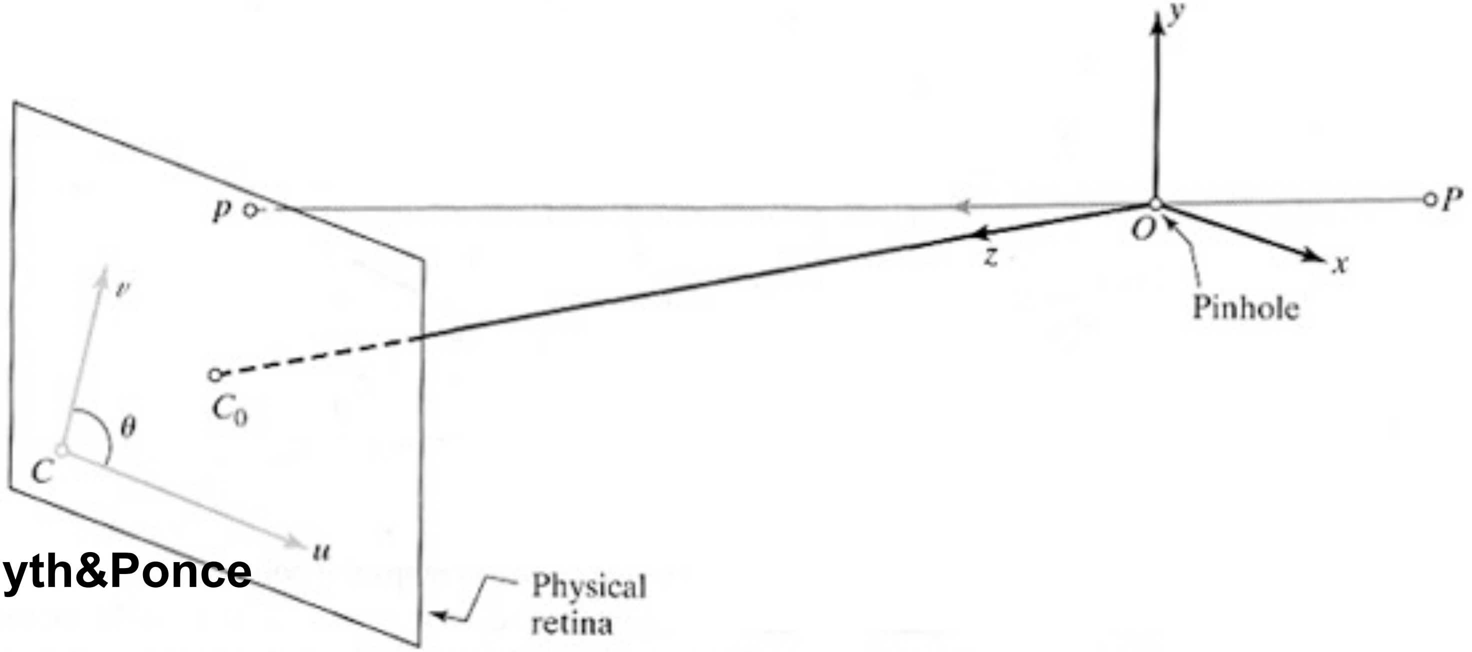
Homogeneous coordinates

$$ {}^{B}\vec{p} = {}^{B}_{A}C\;{}^{A}\vec{p} $$

where

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

Intrinsic parameters: from idealized world coordinates to pixel values



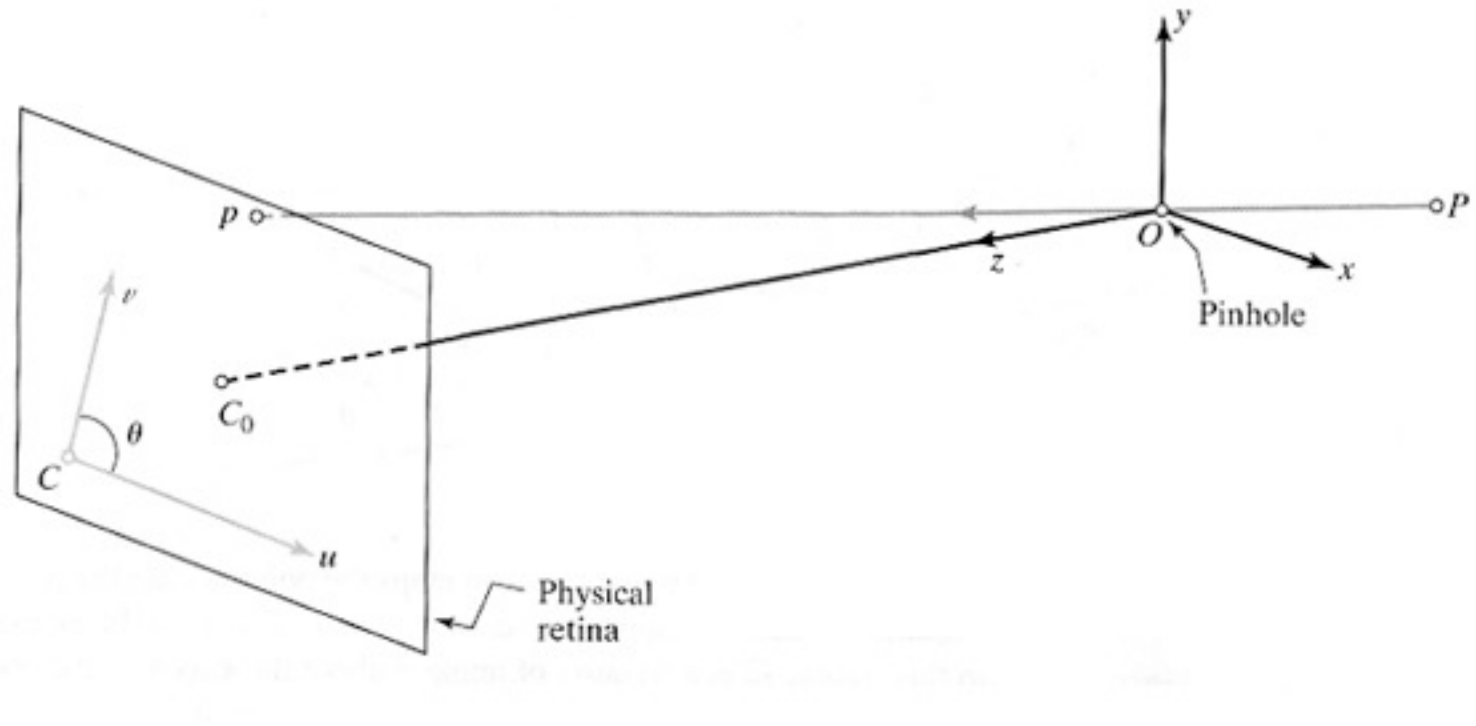
Forsyth&Ponce

Perspective projection

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

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

Intrinsic parameters

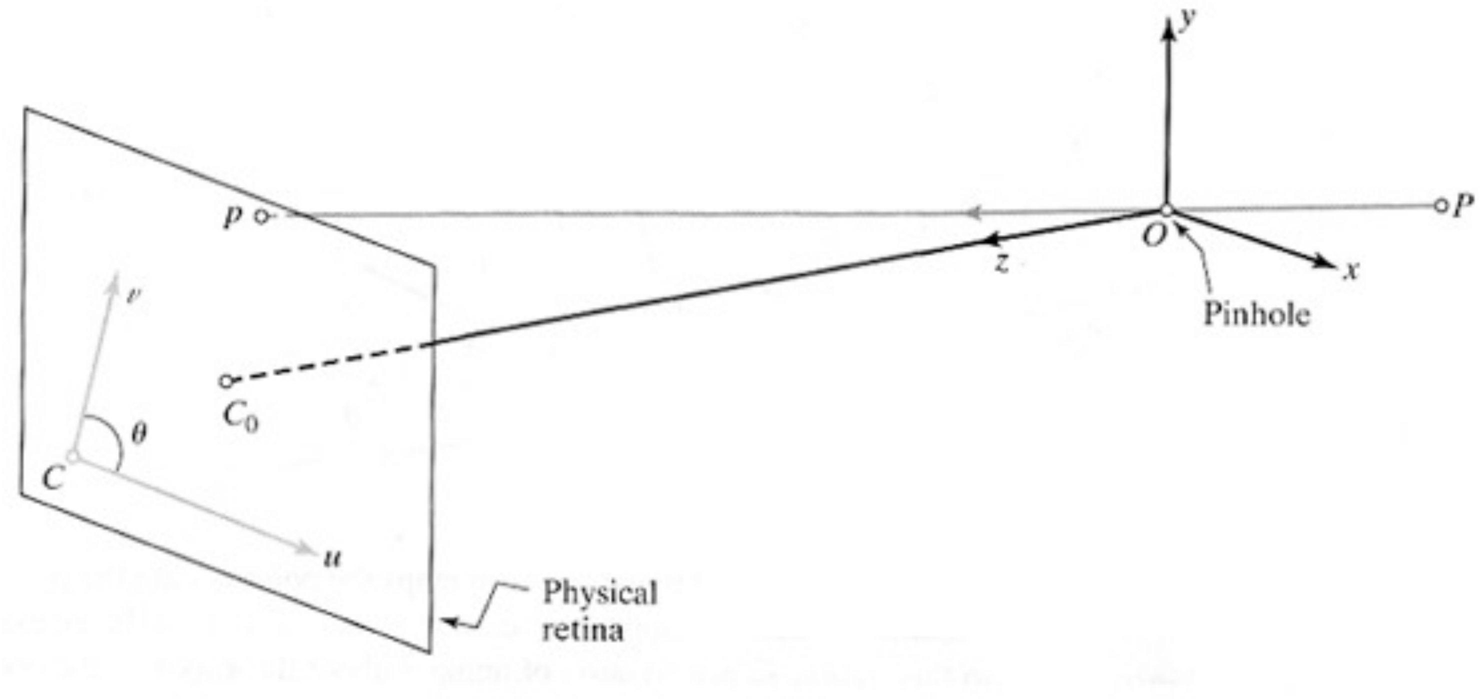


But "pixels" are in some arbitrary spatial units

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

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

Intrinsic parameters

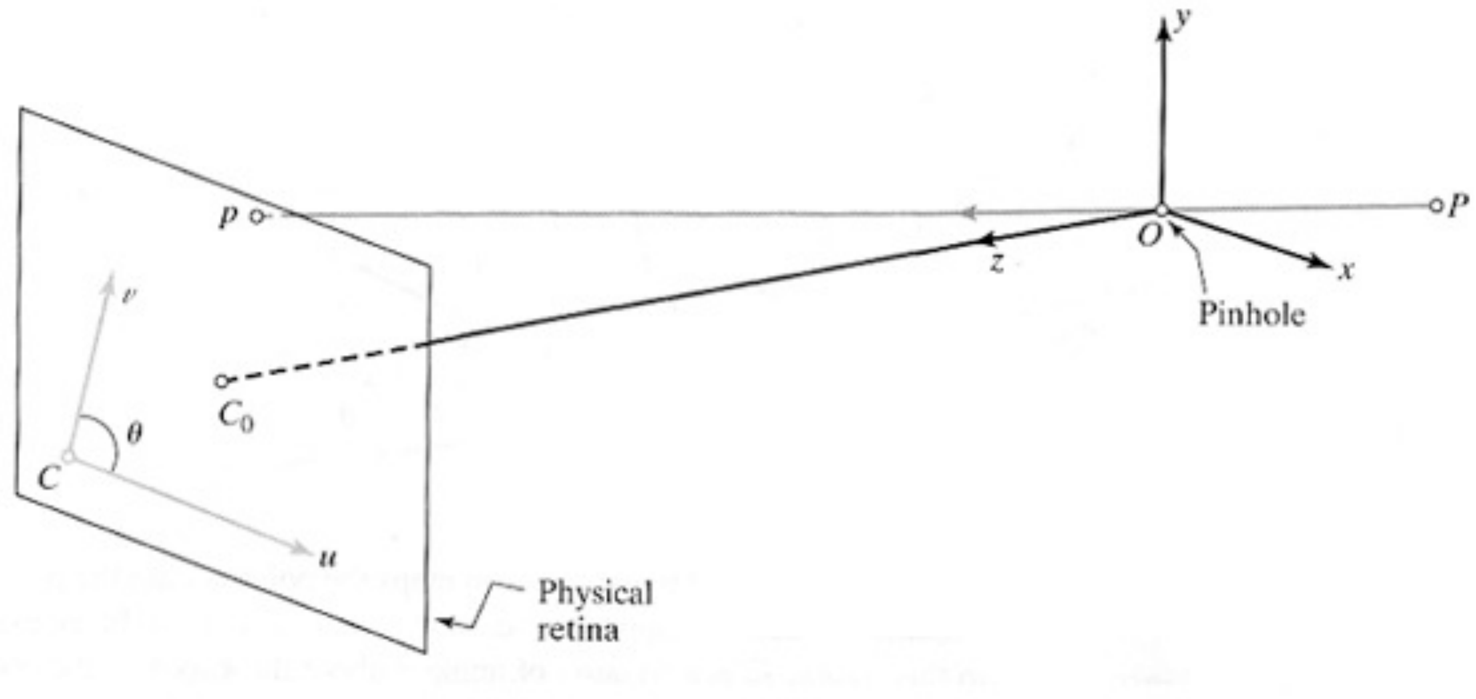


Maybe pixels are not square

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

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

Intrinsic parameters

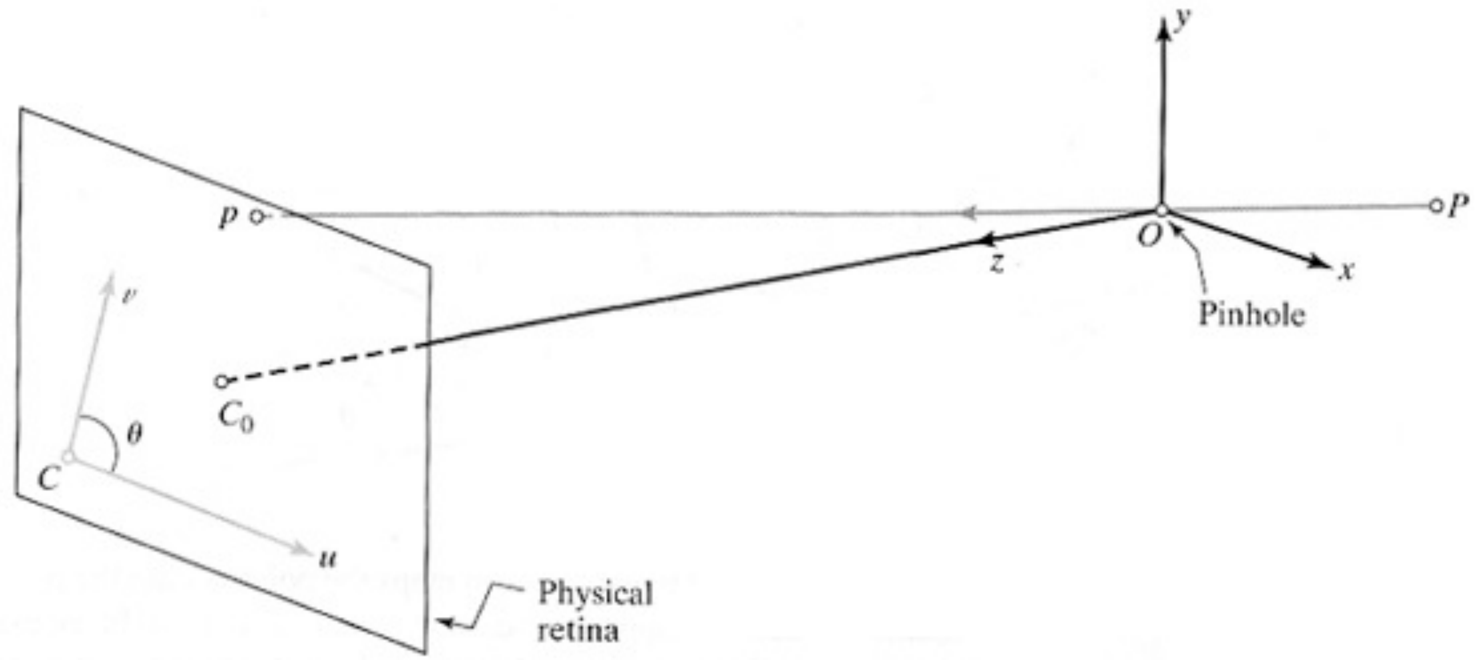


We don't know the origin of our camera pixel coordinates

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

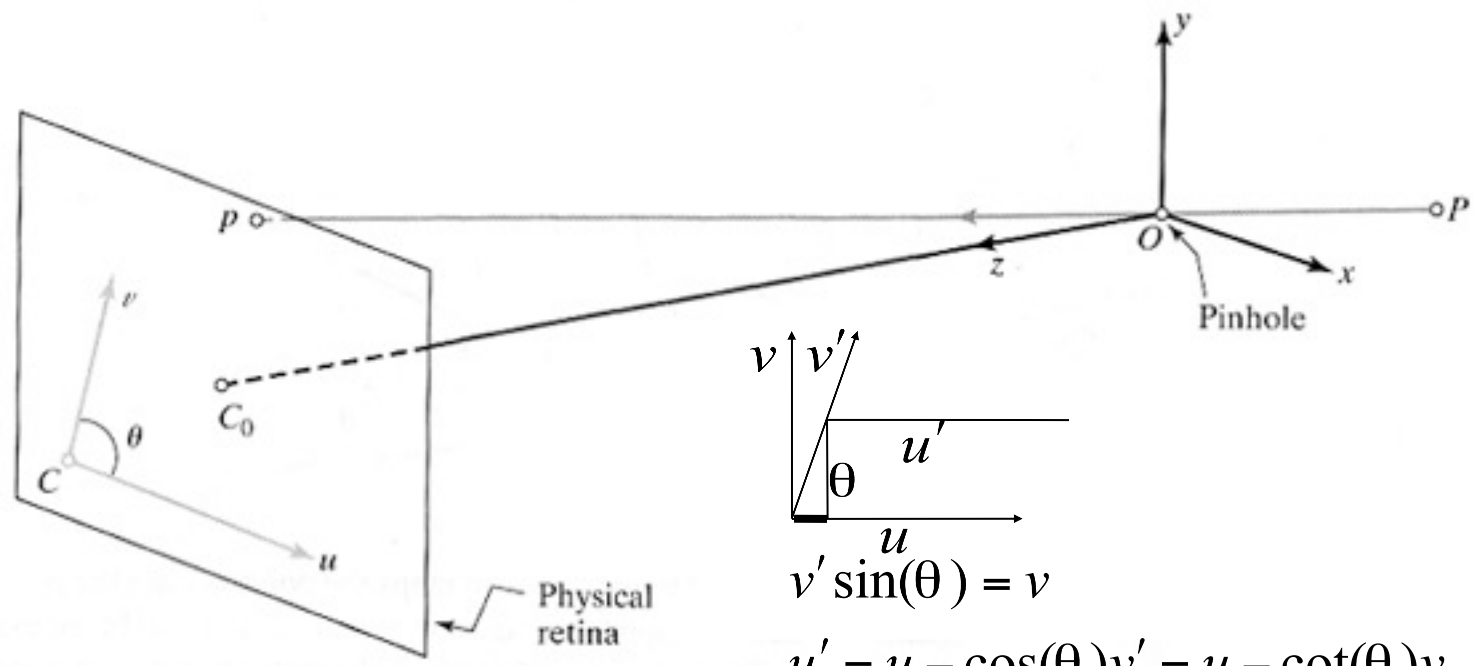
$$v = \beta \frac{y}{z} + v_0$$

Intrinsic parameters



May be skew between camera pixel axes

Intrinsic parameters

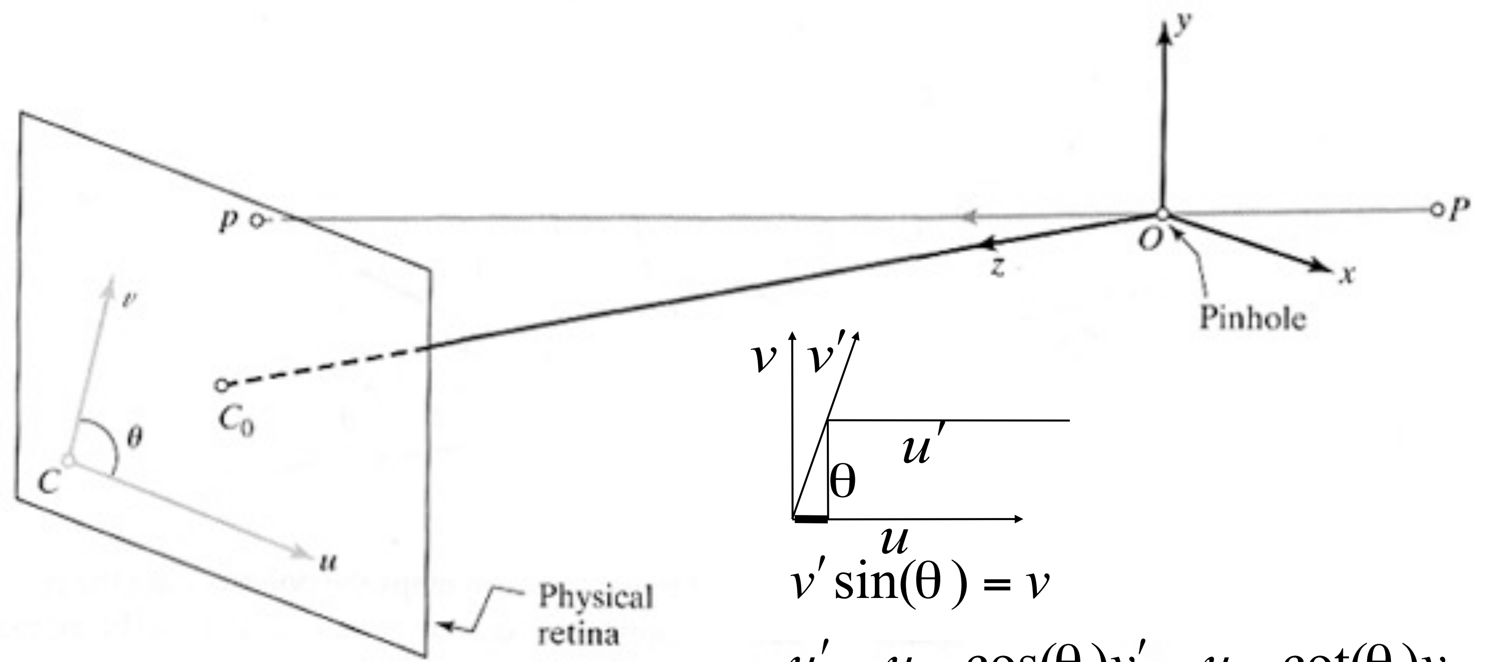


$$v' \sin(\theta) = v$$

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

May be skew between camera pixel axes

Intrinsic parameters



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

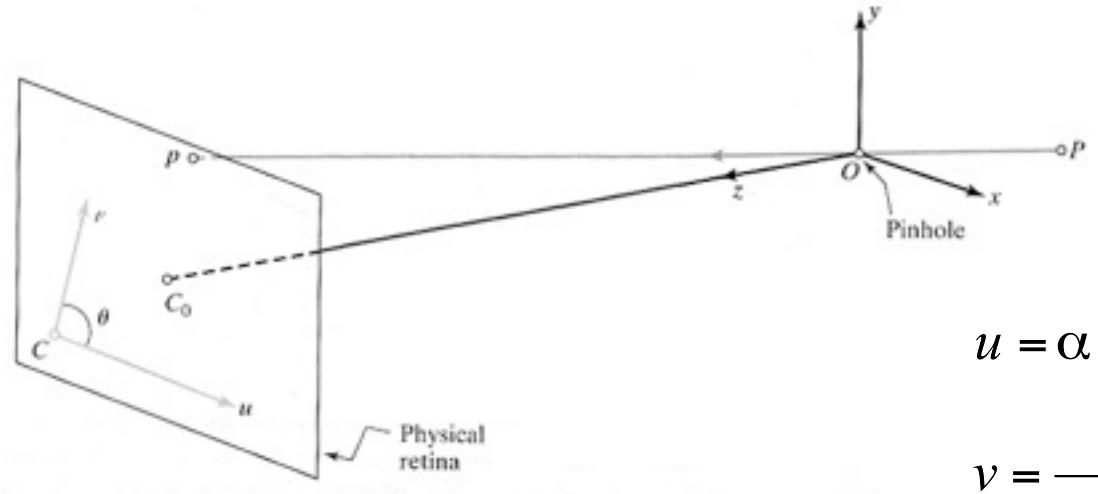
$$u' = u - \cos(\theta)v' = u - \cot(\theta)v$$

May be skew between camera pixel axes

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

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

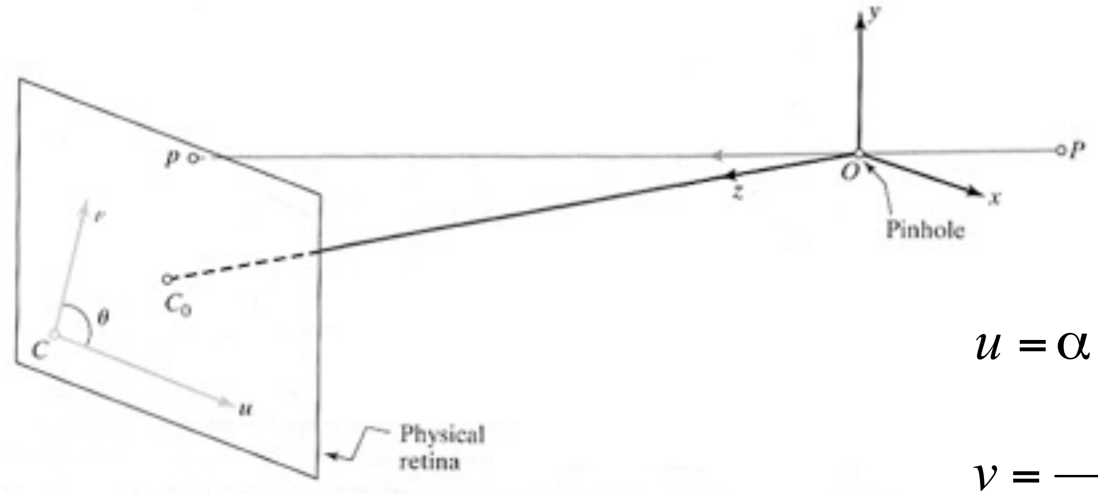
Intrinsic parameters, homogeneous coordinates



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

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

Intrinsic parameters, homogeneous coordinates

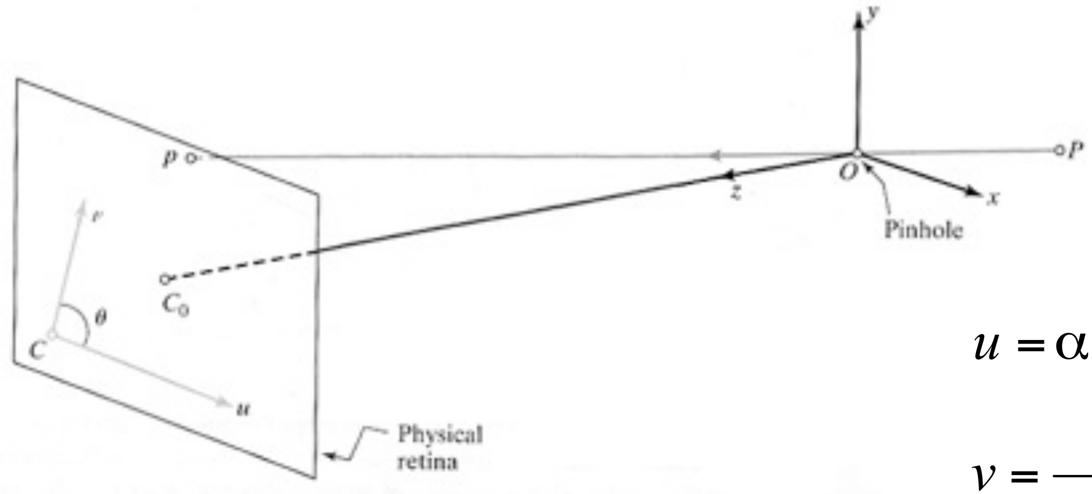


$$u = \alpha \frac{x}{z} - \alpha \cot(\theta) \frac{y}{z} + u_0$$

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

Using homogenous coordinates, we can write this as:

Intrinsic parameters, homogeneous coordinates



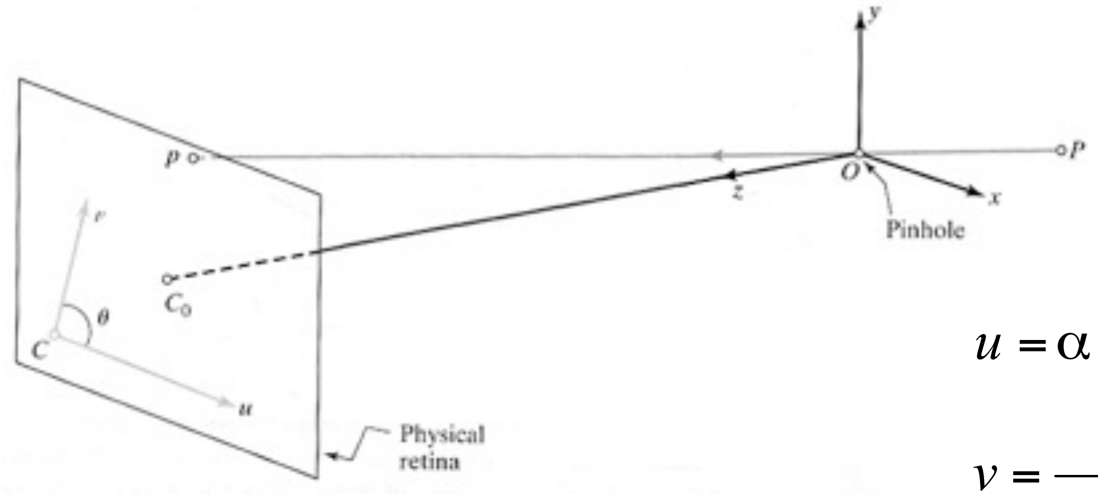
$$u = \alpha \frac{x}{z} - \alpha \cot(\theta)\frac{y}{z} + u_0$$

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

Using homogenous coordinates, we can write this as:

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

Intrinsic parameters, homogeneous coordinates



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

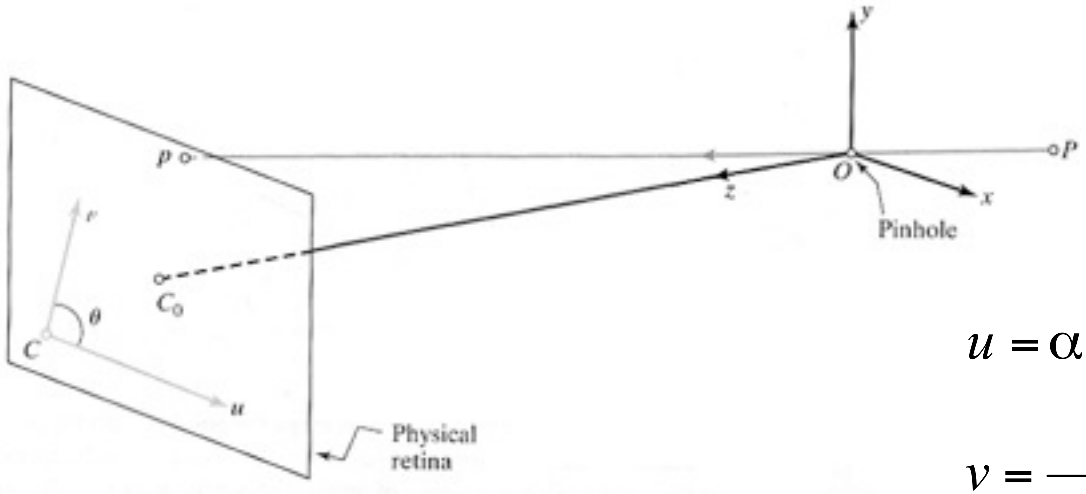
$$v = \frac{\beta}{\sin(\theta)} \frac{y}{z} + v_0$$

Using homogenous coordinates,
we can write this as:

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

or:

Intrinsic parameters, homogeneous coordinates



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

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

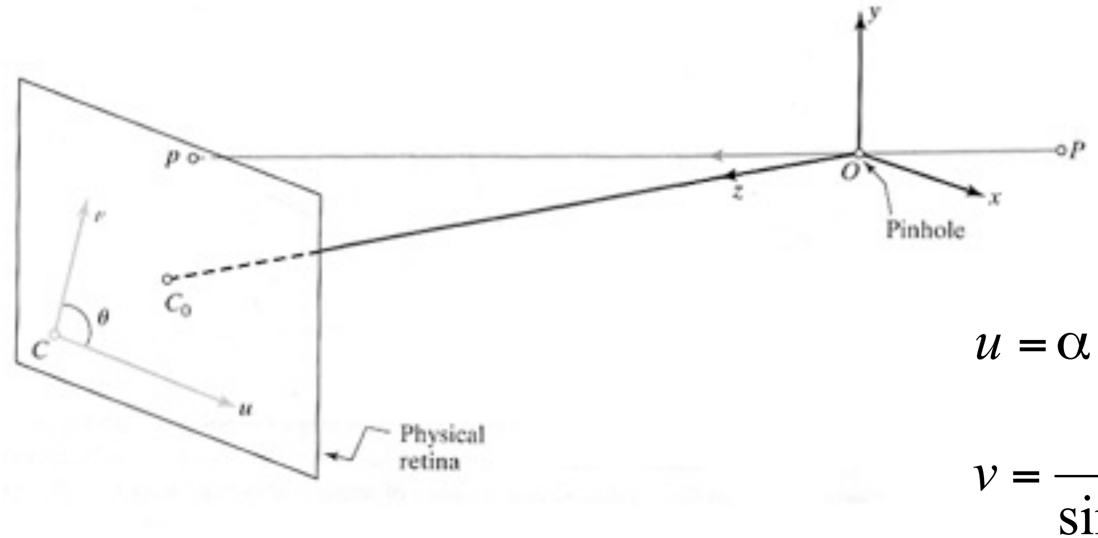
Using homogenous coordinates, we can write this as:

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

or:

$$\vec{p} = \mathrm{K} \; {}^{C}\vec{p}$$

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 →

$$\vec{p} = \mathrm{K} \; {}^{C}\vec{p}$$

In camera-based coords

Extrinsic parameters: translation and rotation of camera frame

Extrinsic parameters: translation and rotation of camera frame

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

Non-homogeneous coordinates

Extrinsic parameters: translation and rotation of camera frame

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

Non-homogeneous coordinates

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

Homogeneous coordinates

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates

Intrinsic

$$\vec{p} = \mathrm{K}\ {}^{C}\vec{p}$$

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

Extrinsic

Forsyth&Ponce

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates

Intrinsic

$$\vec{p} = \mathrm{K}\ {}^{C}\vec{p}$$

World coordinate

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

Extrinsic

Forsyth&Ponce

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates

Intrinsic

$$\vec{p} = \mathrm{K}\ {}^{C}\vec{p}$$

World coordinate

Camera coordinates

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

Extrinsic

Forsyth&Ponce

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates

pixels

Intrinsic

$$\vec{p} = \mathrm{K}\ {}^{C}\vec{p}$$

World coordinate

Camera coordinates

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

Extrinsic

Forsyth&Ponce

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates

pixels

Intrinsic

$$\vec{p} \;=\; \mathrm{K}\;\, {}^{C}\vec{p}$$

World coordinate

Camera coordinates

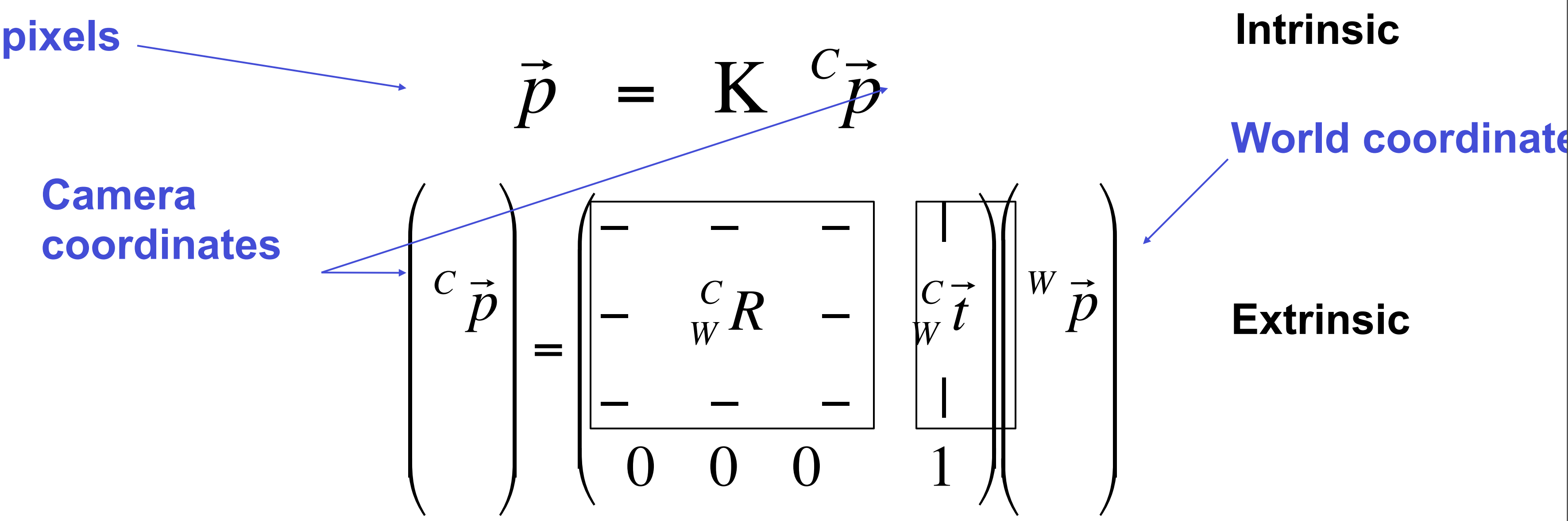
$$\begin{pmatrix} {}^{C}\vec{p} \\ \\ \end{pmatrix} = \begin{pmatrix} \boxed{\begin{matrix} - & - & - \\ - & {}^{C}_{W}R & - \\ - & - & - \end{matrix}} & \boxed{\begin{matrix} | \\ {}^{C}_{W}\vec{t} \\ | \end{matrix}} \\ 0 \quad 0 \quad 0 & 1 \end{pmatrix} \begin{pmatrix} {}^{W}\vec{p} \\ \\ \end{pmatrix}$$

Extrinsic

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

Forsyth&Ponce

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates



pixels

Intrinsic

$$\vec{p} = \mathrm{K}\ {}^{C}\vec{p}$$

World coordinate

Camera coordinates

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

Extrinsic

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

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

Forsyth&Ponce

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates

pixels

Intrinsic

$$\vec{p} = K\ {}^{C}\vec{p}$$

Camera coordinates

World coordinate

Extrinsic

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

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

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

Forsyth&Ponce

Combining extrinsic and intrinsic calibration parameters, in homogeneous coordinates

pixels

Intrinsic

$$\vec{p} \;=\; \mathrm{K} \;\, {}^{C}\vec{p}$$

Camera coordinates

World coordinat

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

Extrinsic

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

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

Forsyth&Ponce

Other ways to write the same equation

pixel coordinates

world coordinates

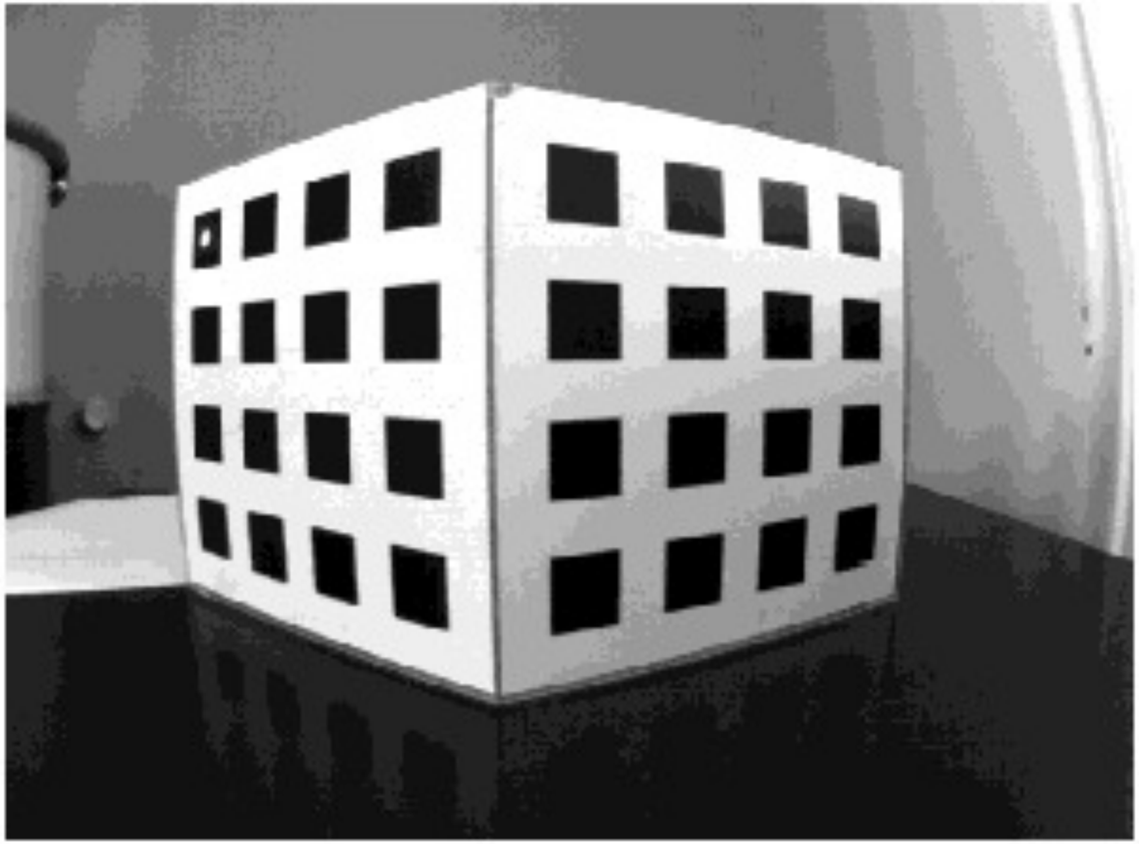
$$\vec{p} = M\ {}^{W}\vec{p}$$

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

Conversion back from homogeneous coordinates leads to:

$$\left\{ \begin{aligned} u &= \frac{m_1^T \vec{P}}{m_3^T \vec{P}} \\ v &= \frac{m_2^T \vec{P}}{m_3^T \vec{P}} \end{aligned} \right.$$

Calibration target



The Opti-CAL Calibration Target Image

Find the position, u_i and v_i, in pixels,
of each calibration object feature point.

http://www.kinetic.bc.ca/CompVision/opti-CAL.html

Camera calibration

Camera calibration

From before, we had these equations relating image positions, u,v, to points at 3-d positions P (in homogeneous coordinates):

$$u = \frac{m_1 \times \vec{P}}{m_3 \times \vec{P}}$$

$$v = \frac{m_2 \times \vec{P}}{m_3 \times \vec{P}}$$

Camera calibration

From before, we had these equations relating image positions, u,v, to points at 3-d positions P (in homogeneous coordinates):

$$u = \frac{m_1 \times \vec{P}}{m_3 \times \vec{P}}$$

$$v = \frac{m_2 \times \vec{P}}{m_3 \times \vec{P}}$$

So for each feature point, i, we have:

$$(m_1 - u_i m_3) \times \vec{P}_i = 0$$

$$(m_2 - v_i m_3) \times \vec{P}_i = 0$$

Camera calibration

Stack all these measurements of i=1…n points

$$(m_1 - u_i m_3) \times \vec{P}_i = 0$$

$$(m_2 - v_i m_3) \times \vec{P}_i = 0$$

into a big matrix:

Camera calibration

Stack all these measurements of i=1…n points

$$(m_1 - u_i m_3) \times \vec{P}_i = 0$$

$$(m_2 - v_i m_3) \times \vec{P}_i = 0$$

into a big matrix:

$$\begin{pmatrix} P_1^T & 0^T & -u_1 P_1^T \\ 0^T & P_1^T & -v_1 P_1^T \\ \cdots & \cdots & \cdots \\ P_n^T & 0^T & -u_n P_n^T \\ 0^T & P_n^T & -v_n P_n^T \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Camera calibration

In vector form:

$$\begin{pmatrix} P_1^T & 0^T & -u_1P_1^T \\ 0^T & P_1^T & -v_1P_1^T \\ \cdots & \cdots & \cdots \\ P_n^T & 0^T & -u_nP_n^T \\ 0^T & P_n^T & -v_nP_n^T \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Showing all the elements:

$$\begin{pmatrix} P_{1x} & P_{1y} & P_{1z} & 1 & 0 & 0 & 0 & 0 & -u_1P_{1x} & -u_1P_{1y} & -u_1P_{1z} & -u_1 \\ 0 & 0 & 0 & 0 & P_{1x} & P_{1y} & P_{1z} & 1 & -v_1P_{1x} & -v_1P_{1y} & -v_1P_{1z} & -v_1 \\ & & & & & & \cdots & \cdots & \cdots & & & \\ P_{nx} & P_{ny} & P_{nz} & 1 & 0 & 0 & 0 & 0 & -u_nP_{nx} & -u_nP_{ny} & -u_nP_{nz} & -u_n \\ 0 & 0 & 0 & 0 & P_{nx} & P_{ny} & P_{nz} & 1 & -v_nP_{nx} & -v_nP_{ny} & -v_nP_{nz} & -v_n \end{pmatrix} \begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Camera calibration

$$\begin{pmatrix} P_{1x} & P_{1y} & P_{1z} & 1 & 0 & 0 & 0 & 0 & -u_1P_{1x} & -u_1P_{1y} & -u_1P_{1z} & -u_1 \\ 0 & 0 & 0 & 0 & P_{1x} & P_{1y} & P_{1z} & 1 & -v_1P_{1x} & -v_1P_{1y} & -v_1P_{1z} & -v_1 \\ & & & & & & \cdots & \cdots & \cdots & & & \\ P_{nx} & P_{ny} & P_{nz} & 1 & 0 & 0 & 0 & 0 & -u_nP_{nx} & -u_nP_{ny} & -u_nP_{nz} & -u_n \\ 0 & 0 & 0 & 0 & P_{nx} & P_{ny} & P_{nz} & 1 & -v_nP_{nx} & -v_nP_{ny} & -v_nP_{nz} & -v_n \end{pmatrix}, \begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix},$$

$$Q \qquad m = 0$$

We want to solve for the unit vector m (the stacked one) that minimizes $|Qm|^2$

The minimum eigenvector of the matrix Q^TQ gives us that (see Forsyth&Ponce, 3.1), because it is the unit vector x that minimizes $x^T Q^TQ \ x$.

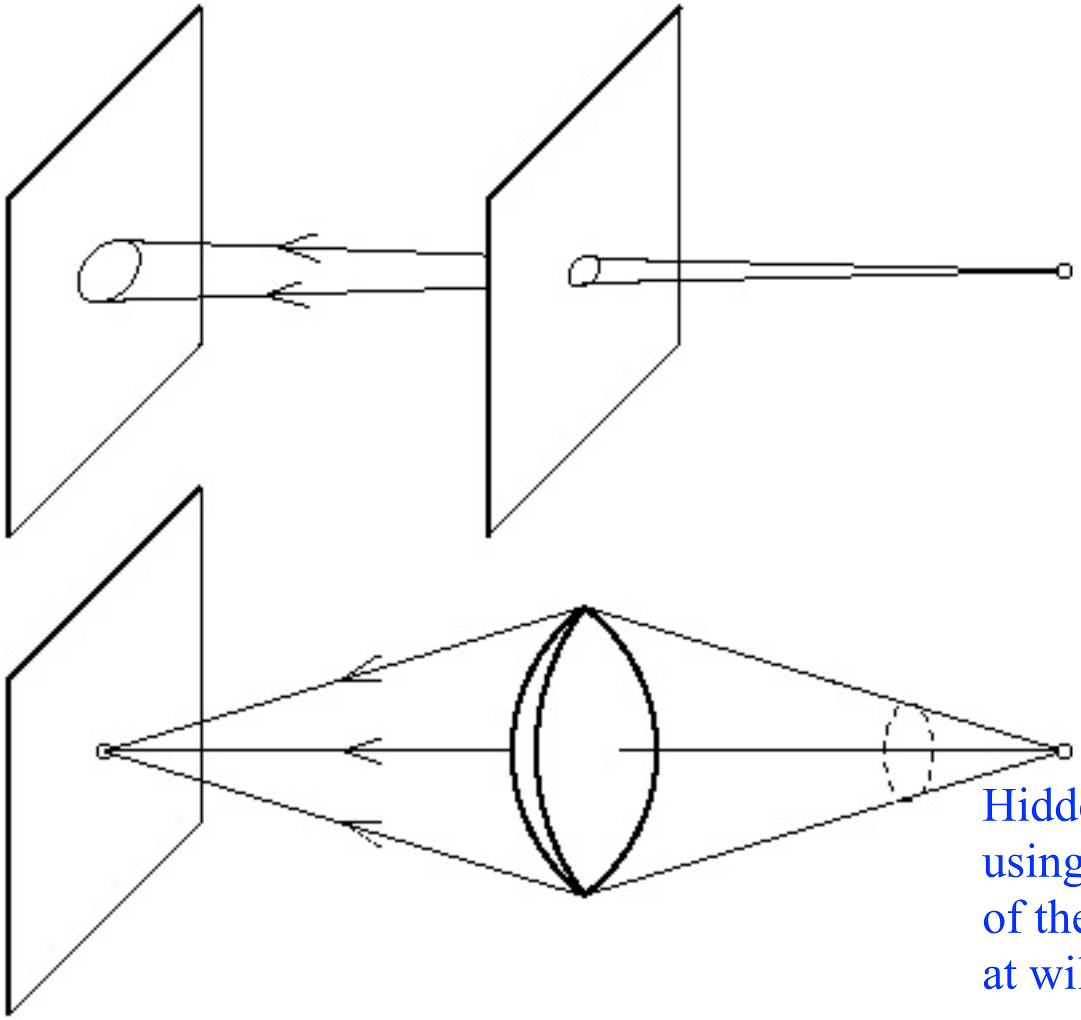
Camera calibration

Once you have the M matrix, can recover the intrinsic and extrinsic parameters.

$$\mathcal{M} = \begin{pmatrix} \alpha \boldsymbol{r}_1^T - \alpha \cot\theta \boldsymbol{r}_2^T + u_0 \boldsymbol{r}_3^T & \alpha t_x - \alpha \cot\theta t_y + u_0 t_z \\ \dfrac{\beta}{\sin\theta} \boldsymbol{r}_2^T + v_0 \boldsymbol{r}_3^T & \dfrac{\beta}{\sin\theta} t_y + v_0 t_z \\ \boldsymbol{r}_3^T & t_z \end{pmatrix}$$

Why do we need lenses?

The reason for lenses: light gathering ability



Hidden assumption in using a lens: the BRDF of the surfaces you look at will be well-behaved

Animal Eyes

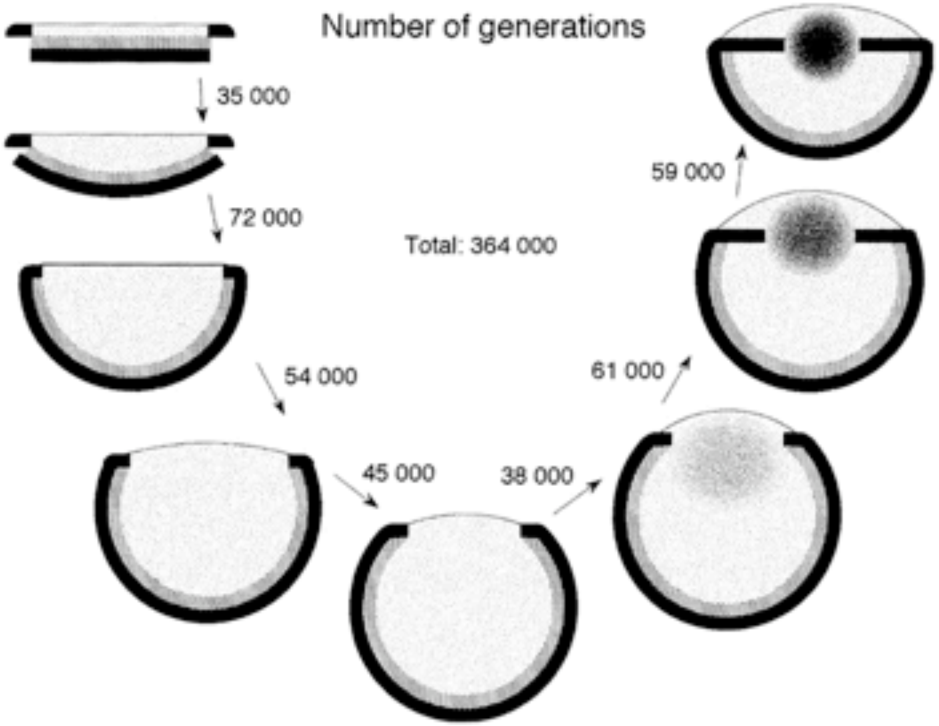


Fig. 1.6 A patch of light sensitive epithelium can be gradually turned into a perfectly focussed camera-type eye if there is a continuous selection for improved spatial vision. A theoretical model based on conservative assumptions about selection pressure and the amount of variation in natural populations suggest that the whole sequence can be accomplished amazingly fast, in less than 400 000 generations. The number of generations is also given between each of the consecutive intermediates that are drawn in the figure. The starting point is a flat piece of epithelium with an outer protective layer, an intermediate layer of receptor cells, and a bottom layer of pigment cells. The first half of the sequence is the formation of a pigment cup eye. When this principle cannot be improved any further, a lens gradually evolves. Modified from Nilsson and Pelger (1994).

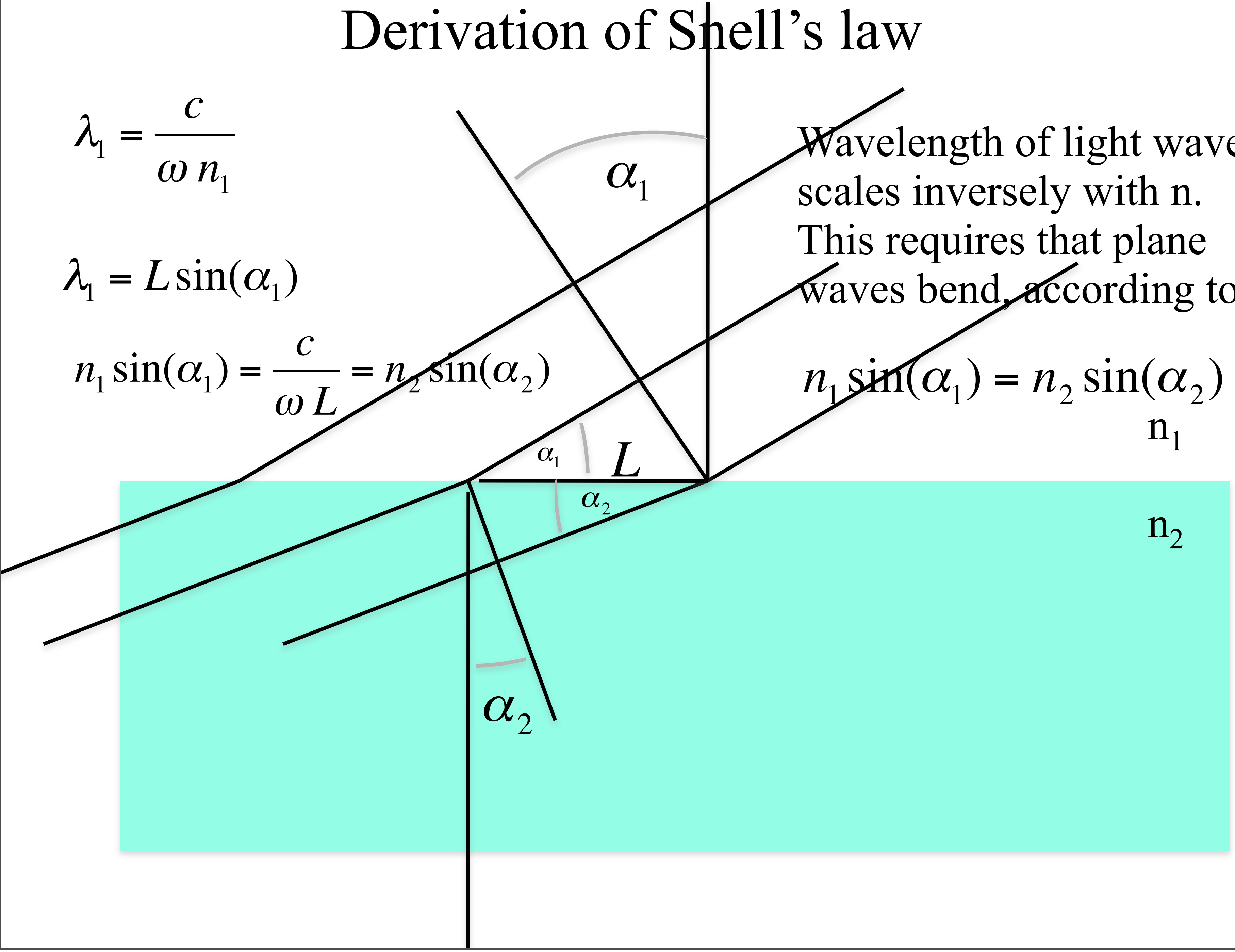
Animal Eyes. Land & Nilsson. Oxford Univ. Press

Derivation of Snell’s law

$$\lambda_1 = \frac{c}{\omega\, n_1}$$

$$\lambda_1 = L\sin(\alpha_1)$$

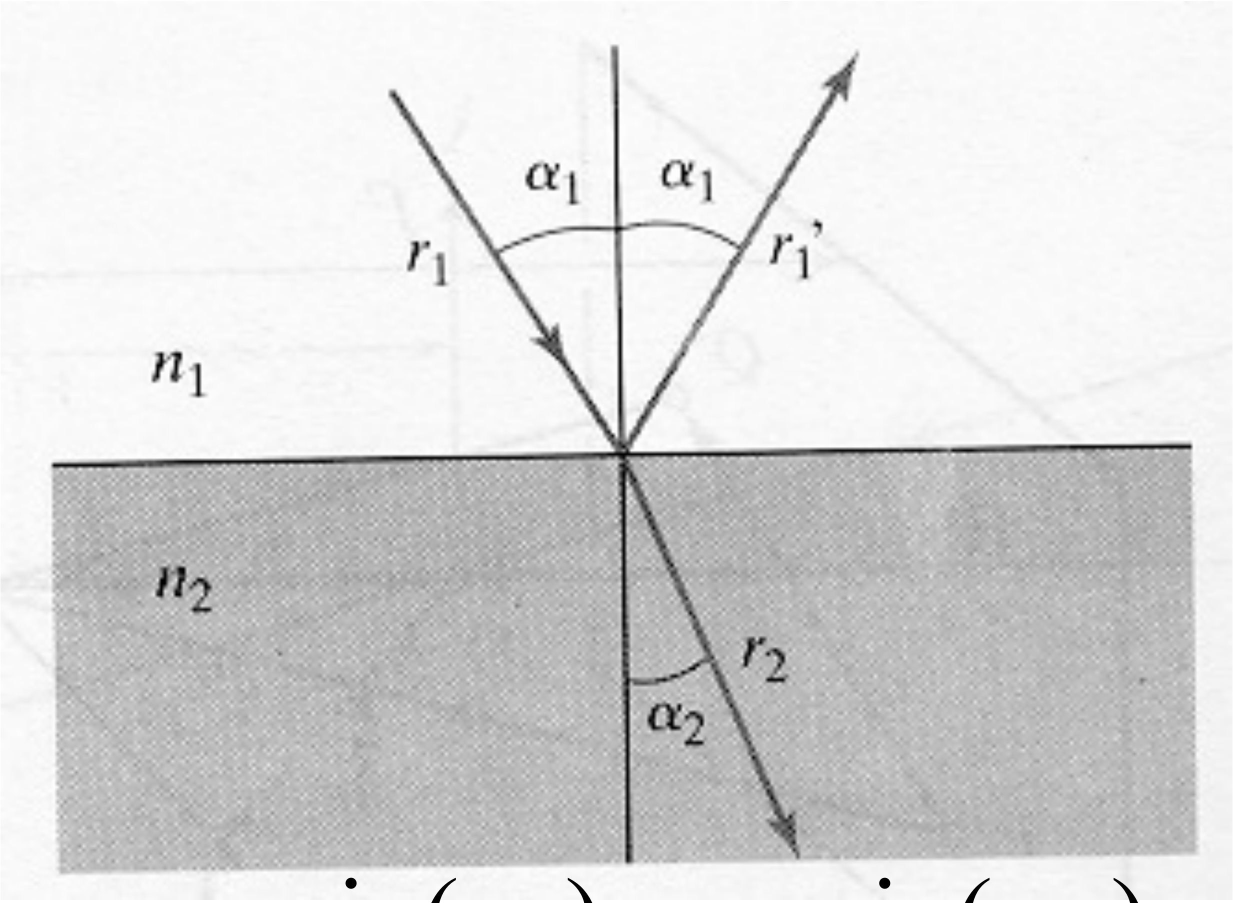
$$n_1 \sin(\alpha_1) = \frac{c}{\omega\, L} = n_2 \sin(\alpha_2)$$



Wavelength of light waves scales inversely with n. This requires that plane waves bend, according to

$$n_1 \sin(\alpha_1) = n_2 \sin(\alpha_2)$$

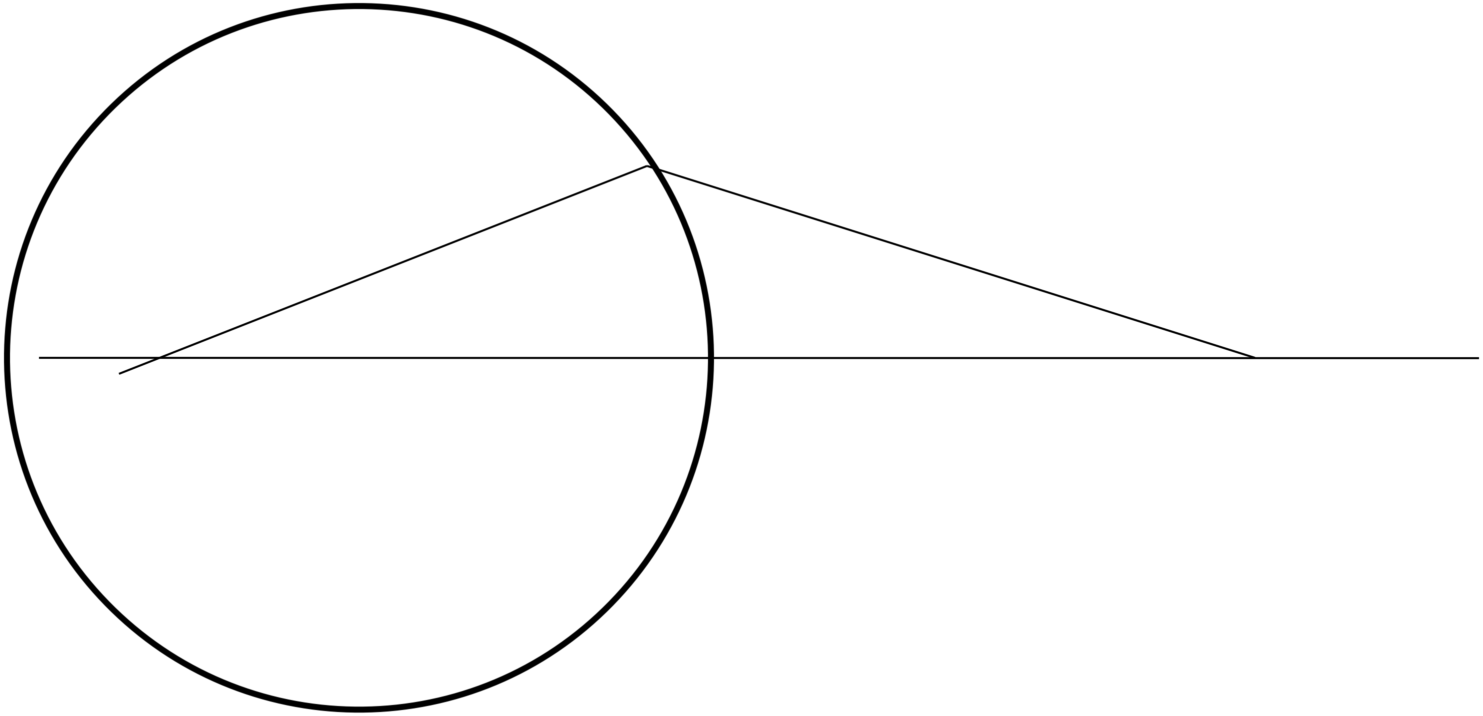
Refraction: Snell's law

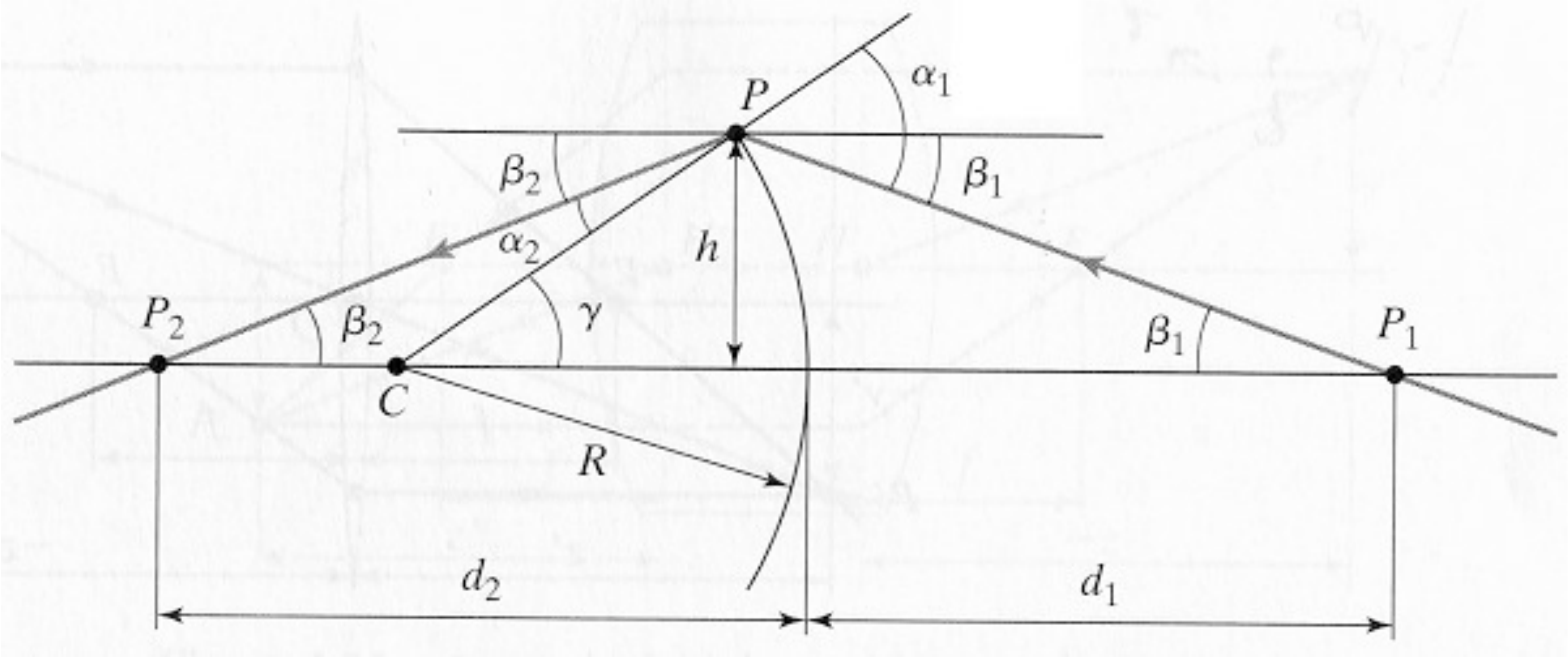


$$n_1 \sin(\alpha_1) = n_2 \sin(\alpha_2)$$

For small angles, $n_1\alpha_1 \approx n_2\alpha_2$

Spherical lens

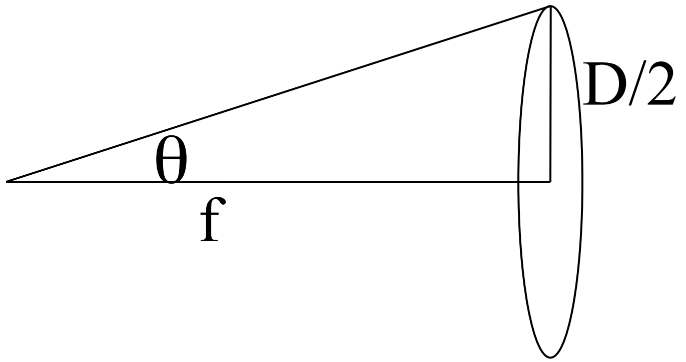




Forsyth and Ponce

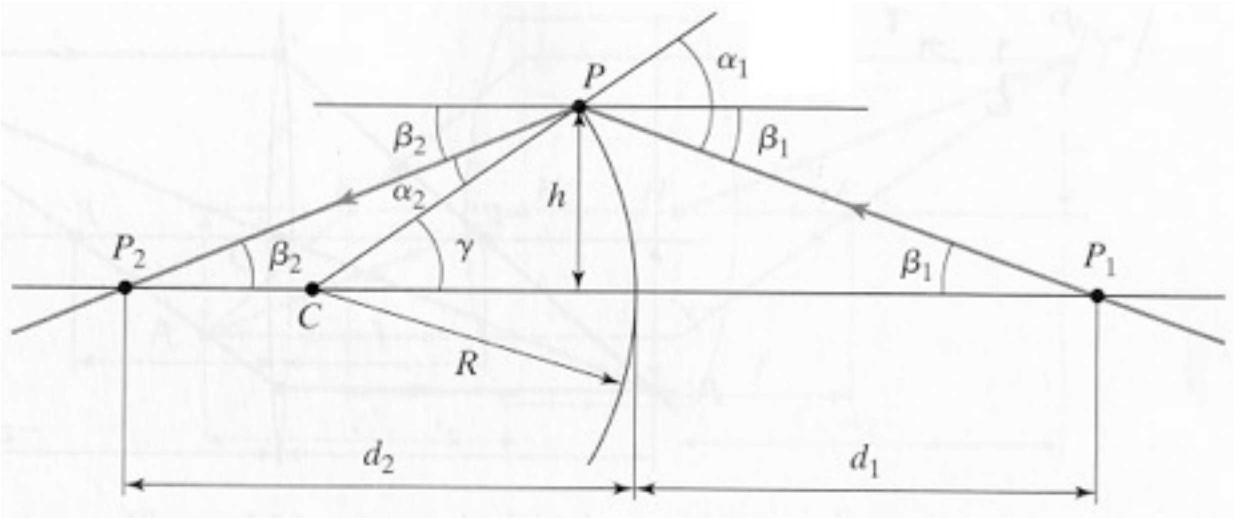
First order optics

$$\sin(\theta) \approx \theta$$



$$\theta \approx \frac{D/2}{f}$$

Paraxial refraction equation

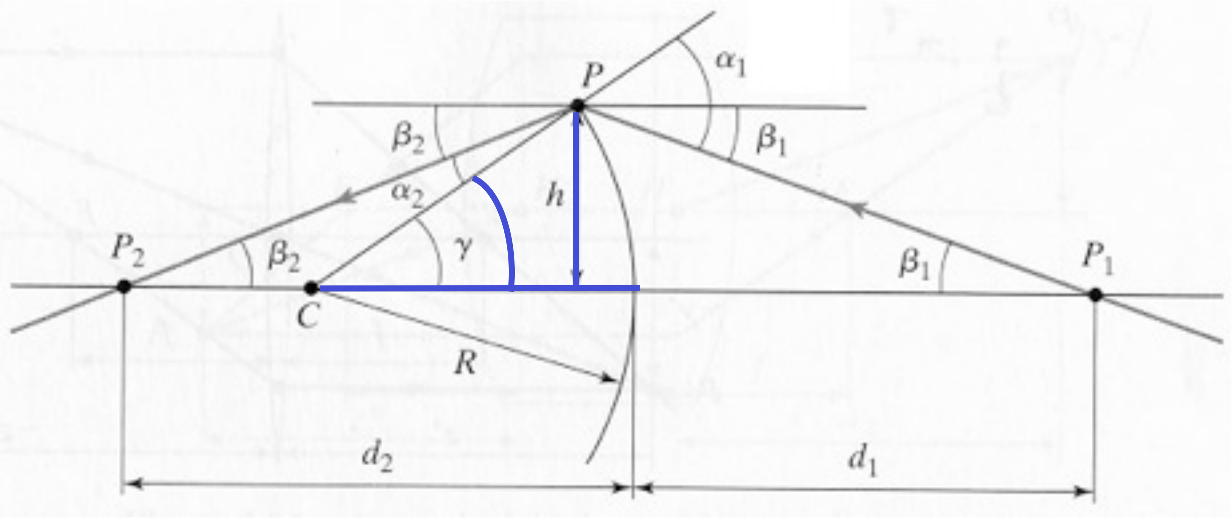


$$\alpha_1 = \gamma + \beta_1 \approx h\left(\frac{1}{R} + \frac{1}{d_1}\right)$$

Relates distance from sphere and sphere radius to bending angles of light ray, for lens with one spherical surface.

$$n_1\alpha_1 \approx n_2\alpha_2$$

Paraxial refraction equation

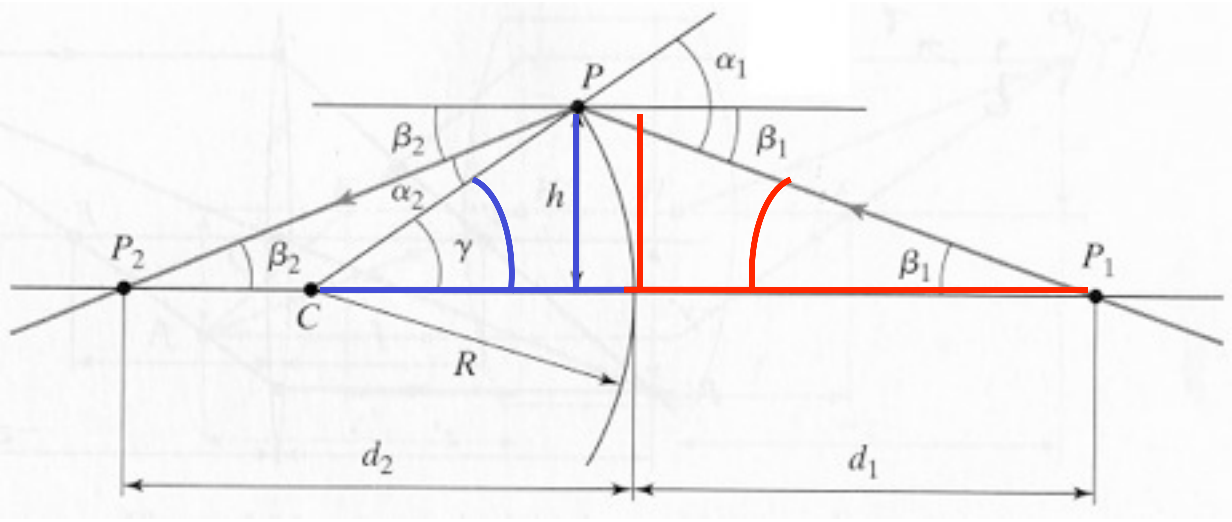


$$\alpha_1 = \boxed{\gamma} + \beta_1 \approx h\left(\frac{1}{R} + \frac{1}{d_1}\right)$$

Relates distance from sphere and sphere radius to bending angles of light ray, for lens with one spherical surface.

$$n_1\alpha_1 \approx n_2\alpha_2$$

Paraxial refraction equation

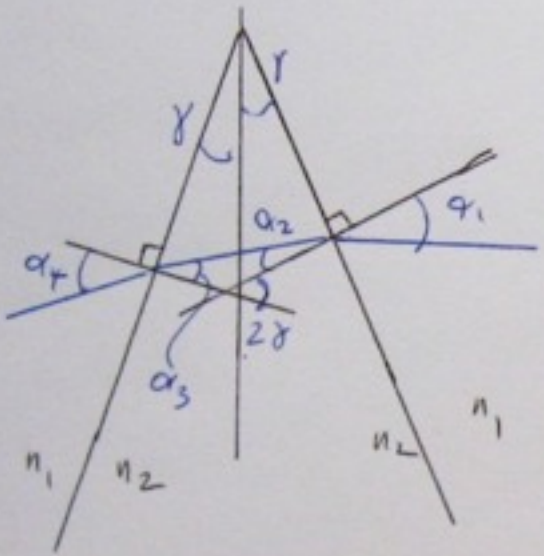


$$\alpha_1 = \boxed{\gamma} + \boxed{\beta_1} \approx h\left(\frac{1}{R}+\frac{1}{d_1}\right)$$

Relates distance from sphere and sphere radius to bending angles of light ray, for lens with one spherical surface.

$$n_1\alpha_1 \approx n_2\alpha_2$$

Deriving the lensmaker's formula



$\alpha_1 = h\left(\frac{1}{R} + \frac{1}{d_1}\right)$ small angle approx

$n_1 \alpha_1 \cong n_2 \alpha_2$ snell's law

$\alpha_2 = 2\gamma - \alpha_3$ geometry

$n_2 \alpha_3 \cong n_1 \alpha_4$ snell's law

$\alpha_4 = h_1\left(\frac{1}{R} + \frac{1}{d_2}\right)$ small angle

$\gamma = \frac{h}{R}$ small angle

$$n_1 \alpha_1 = n_2\left(\frac{2h}{R} - \frac{n_1}{n_2}\alpha_4\right) = h\left(\frac{1}{R} + \frac{1}{d_1}\right)$$

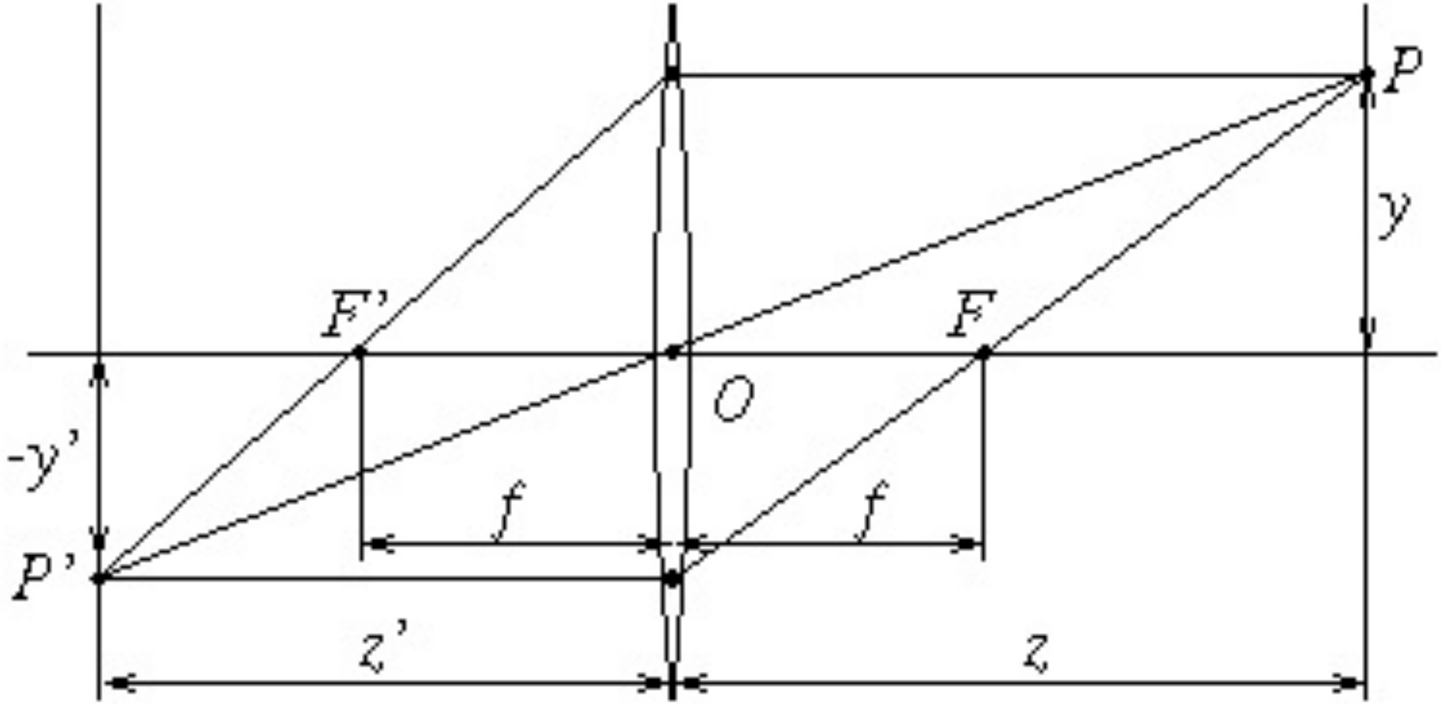
let $n_1 = 1$, $n_2 = n$

cancel h's $n\left(\frac{2}{R} - \frac{1}{n}\left(\frac{1}{R} + \frac{1}{d_2}\right)\right) = \frac{1}{R} + \frac{1}{d_1}$

$$\frac{2n}{R} - \frac{1}{R} - \frac{1}{d_2} = \frac{1}{R} + \frac{1}{d_1}$$

$$\frac{2(n-1)}{R} = \frac{1}{d_1} + \frac{1}{d_2}$$ "Lens maker's formula"

The thin lens, first order optics



The lensmaker's equation:

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

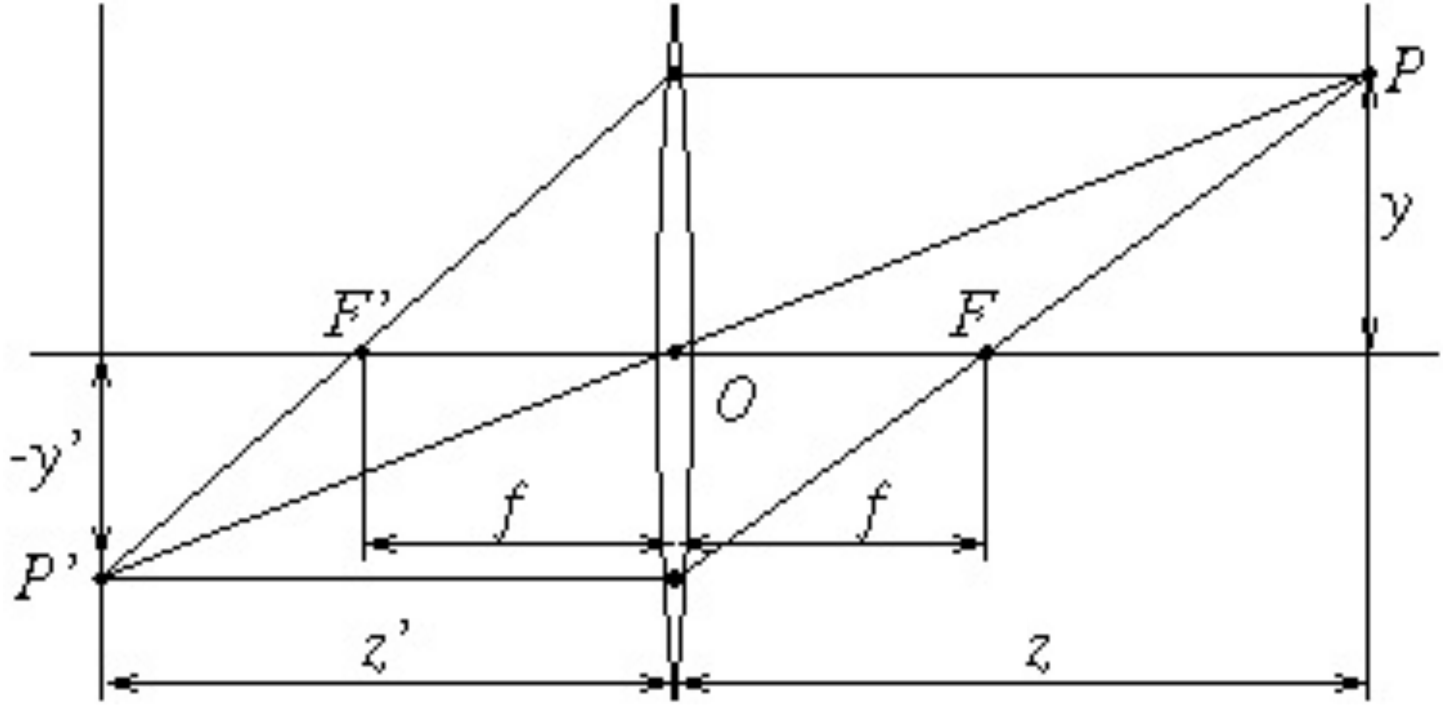
$$f = \frac{R}{2(n-1)}$$

Forsyth&Ponce

What camera projection model applies for a thin lens?

What camera projection model applies for a thin lens?

The perspective projection of a pinhole camera. But note that many more of the rays leaving from P arrive at P’



Lens demonstration

- Verify:
  - Focusing property
  - Lens maker's equation

More accurate models of real lenses

- Finite lens thickness
- Higher order approximation to $\sin(\theta)$
- Chromatic aberration
- Vignetting

Thick lens

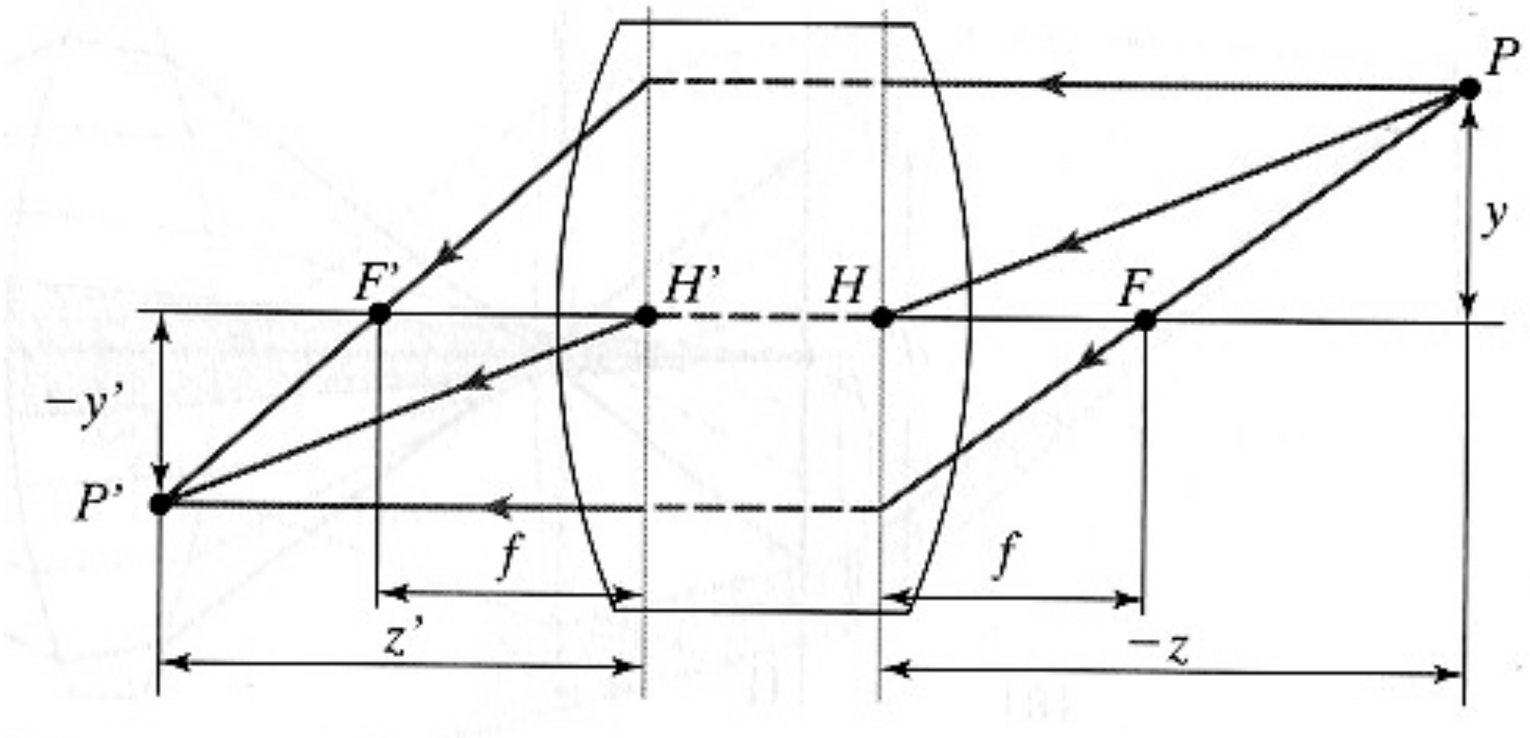
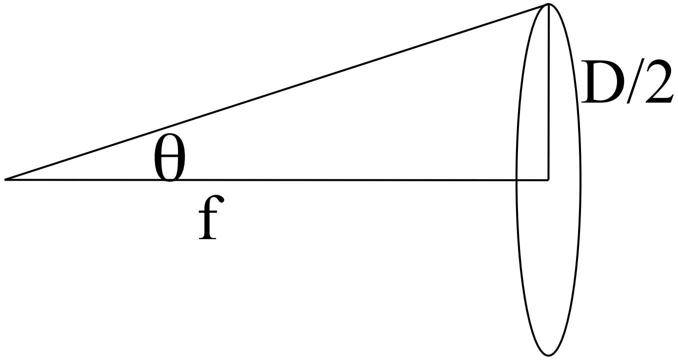


Figure 1.11 A simple thick lens with two spherical surfaces.

Forsyth&Ponce

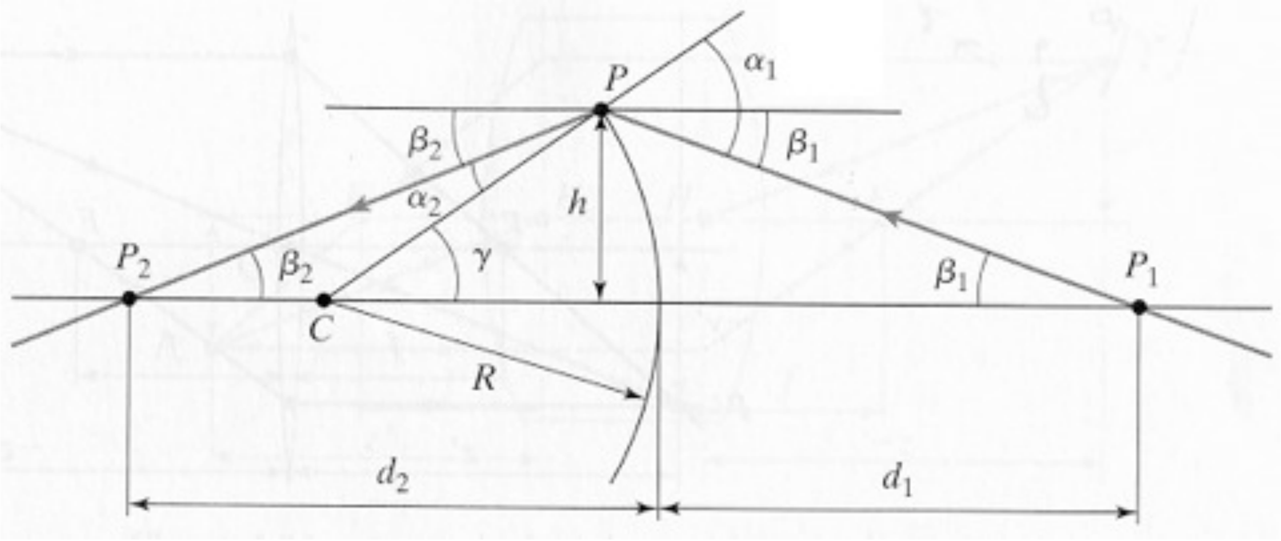
Third order optics

$$\sin(\theta) \approx \theta - \frac{\theta^3}{6}$$



$$\theta \approx \frac{D/2}{f} - \frac{\left(\frac{D/2}{f}\right)^3}{6}$$

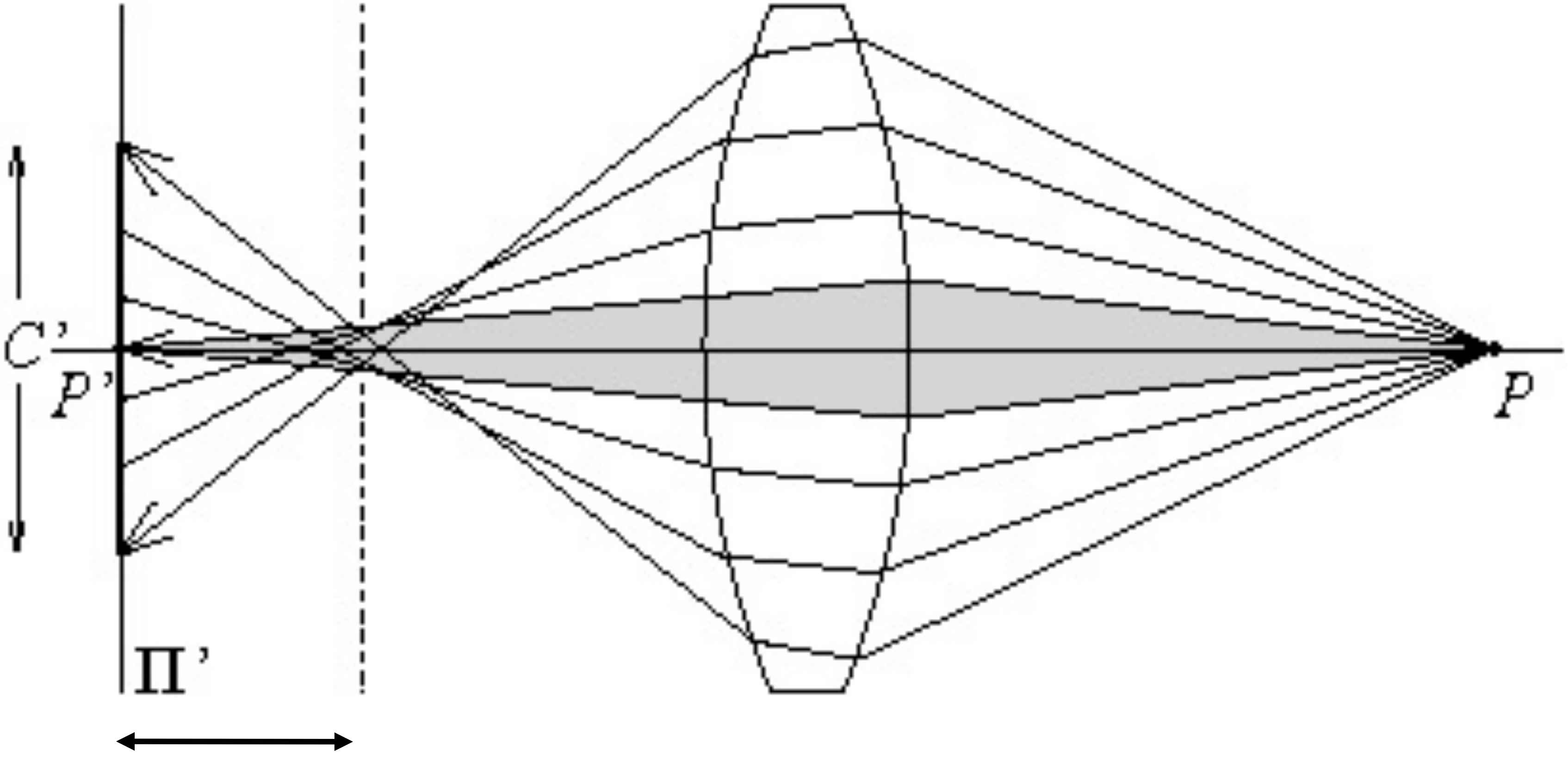
Paraxial refraction equation, 3rd order optics



$$\frac{n_1}{d_1}+\frac{n_2}{d_2}=\frac{n_2-n_1}{R}+h^2\left[\frac{n_1}{2d_1}\left(\frac{1}{R}+\frac{1}{d_1}\right)^2+\frac{n_2}{2d_2}\left(\frac{1}{R}-\frac{1}{d_2}\right)^2\right]$$

Forsyth&Ponce

Spherical aberration (from 3rd order optics



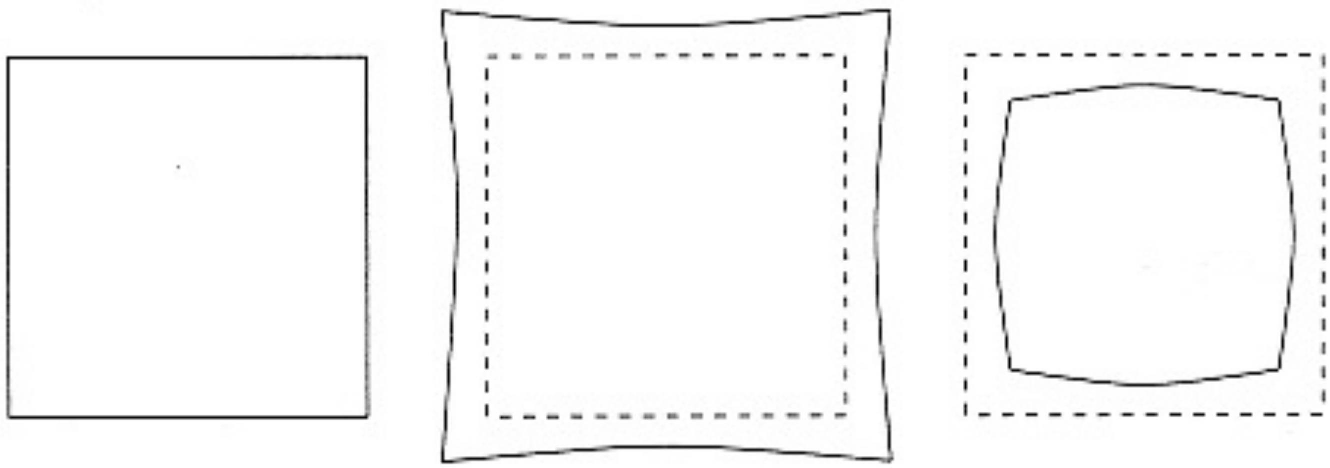
Transverse spherical aberration

Longitudinal spherical aberration

Forsyth&Ponce

Other 3rd order effects

- Coma, astigmatism, field curvature, distortion.



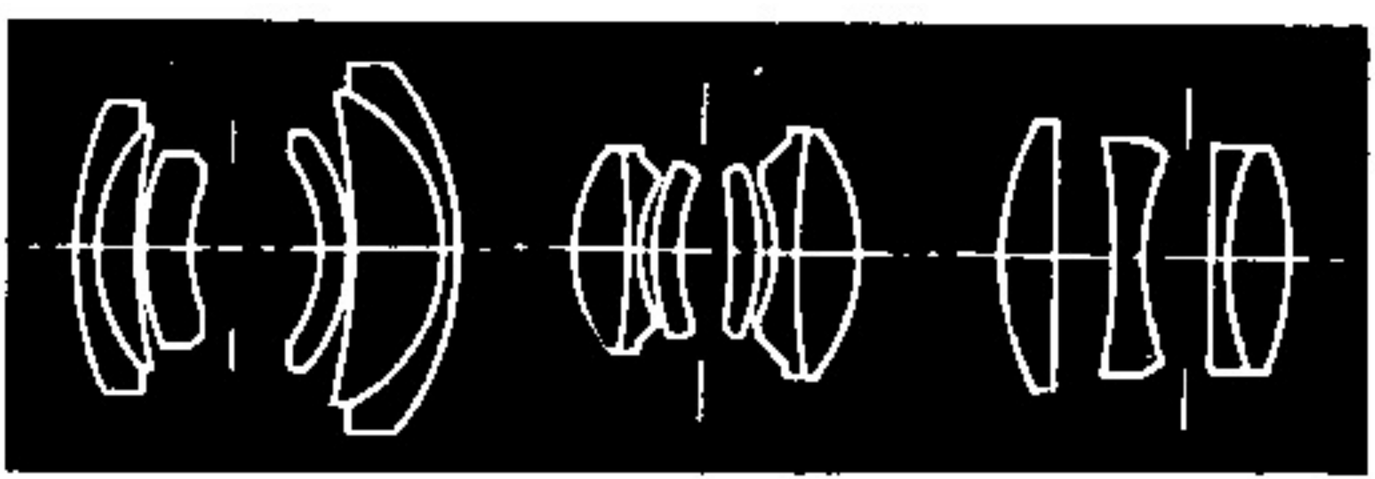
no distortion

pincushion distortion

barrel distortion

Forsyth&Ponce

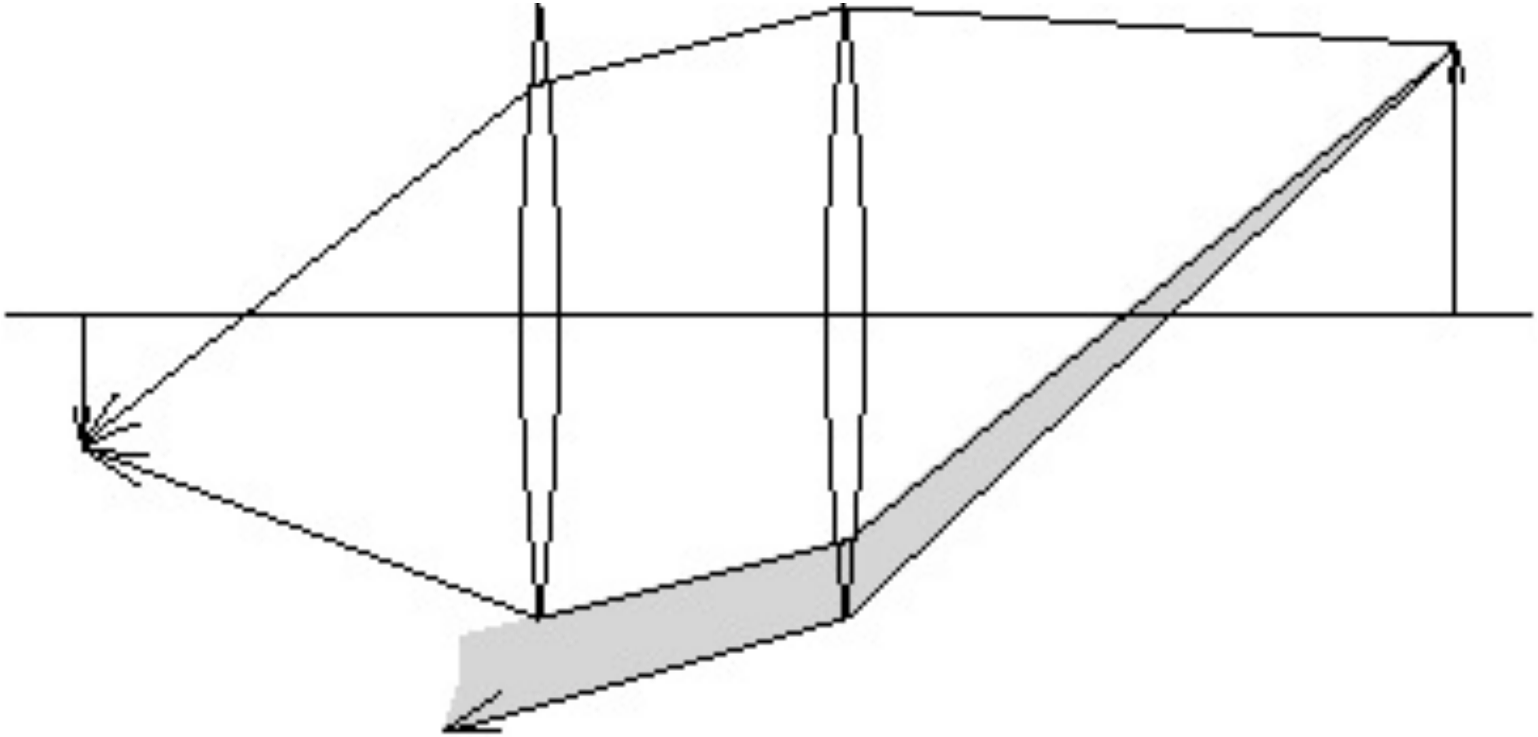
Lens systems



Lens systems can be designed to correct for aberrations described by 3rd order optics

Forsyth&Ponce

Vignetting



Forsyth&Ponce

Chromatic aberration

(desirable for prisms, bad for lenses)

© 1994, The Exploratorium

Other (possibly annoying) phenomena

- Chromatic aberration
  - Light at different wavelengths follows different paths; hence, some wavelengths are defocussed
  - Machines: coat the lens
  - Humans: live with it
- Scattering at the lens surface
  - Some light entering the lens system is reflected off each surface it encounters (Fresnel's law gives details)
  - Machines: coat the lens, interior
  - Humans: live with it (various scattering phenomena are visible in the human eye)

Summary

- Want to make images
- Pinhole camera models the geometry of perspective projection
- Lenses make it work in practice
- Models for lenses
  - Thin lens, spherical surfaces, first order optics
  - Thick lens, higher-order optics, vignetting.