6.891

Computer Vision and Applications

Prof. Trevor. Darrell

- Class overview
- Administrivia & Policies
- Lecture 1
  - Perspective projection (review)
  - Rigid motions (review)
  - Camera Calibration

Readings: Forsythe & Ponce, 1.1, 2.1, 2.2, 2.3, 3.1, 3.2

Vision

- What does it mean, to see? “to know what is where by looking”.
- How to discover from images what is present in the world, where things are, what actions are taking place.

from Marr, 1982

Why study Computer Vision?

- One can “see the future” (and avoid bad things…)!
- Images and movies are everywhere; fast-growing collection of useful applications
  - building representations of the 3D world from pictures
  - automated surveillance (who’s doing what)
  - movie post-processing
  - face finding
- Greater understanding of human vision
- Various deep and attractive scientific mysteries
  - how does object recognition work?

Why study Computer Vision?

- People draw distinctions between what is seen
  - “Object recognition”
  - This could mean “is this a fish or a bicycle?”
  - It could mean “is this George Washington?”
  - It could mean “is this poisonous or not?”
  - It could mean “is this slippery or not?”
  - It could mean “will this support my weight?”
  - Great mystery
    - How to build programs that can draw useful distinctions based on image properties.

Computer vision class, fast-forward

Cameras, lenses, and sensors

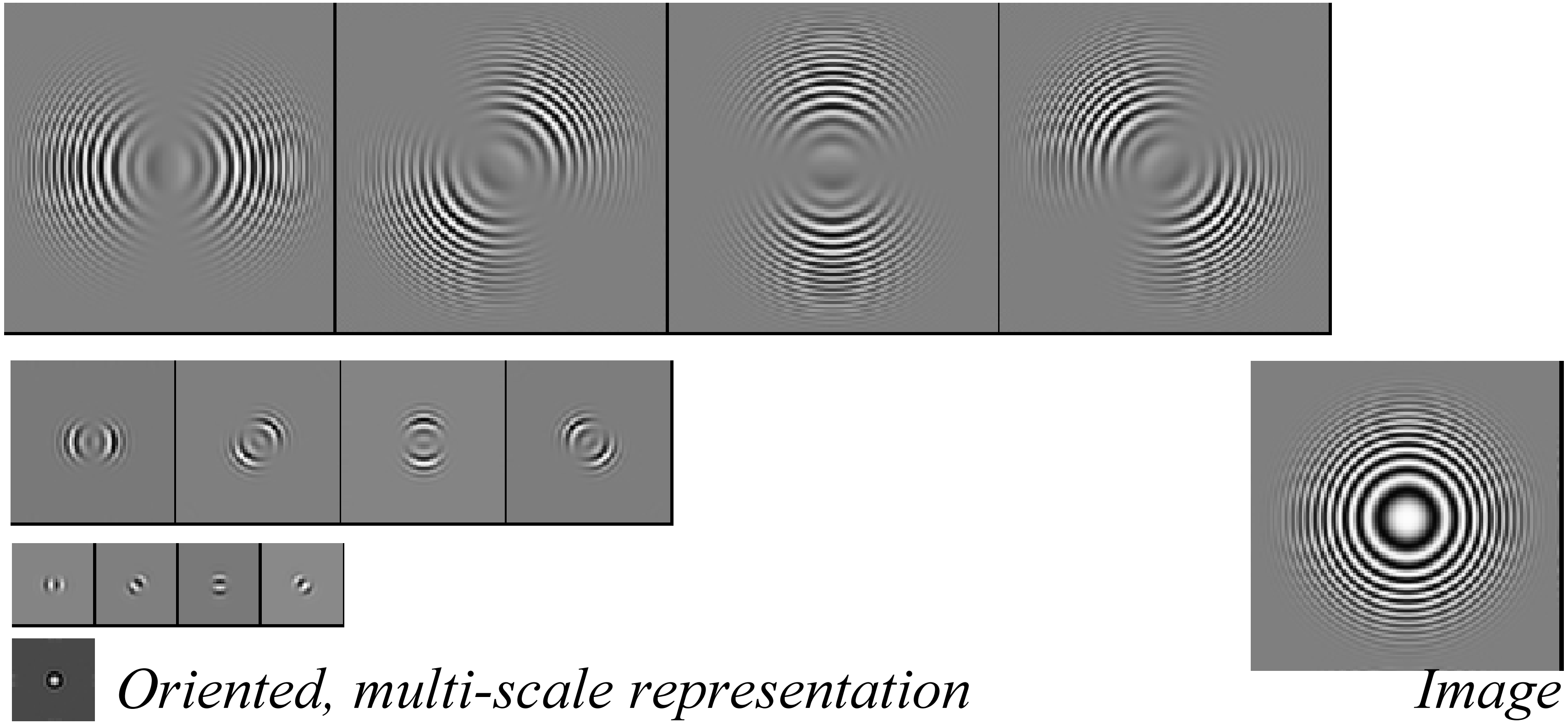
Figure 1.16 The first photograph on record, *la table servie*, obtained by Nicéphore Niepce in 1822. *Collection Harlinge–Viollet.*

- •Pinhole cameras
- •Lenses
- •Projection models
- •Geometric camera parameters

From Computer Vision, Forsyth and Ponce, Prentice-Hall, 2002.

Image filtering

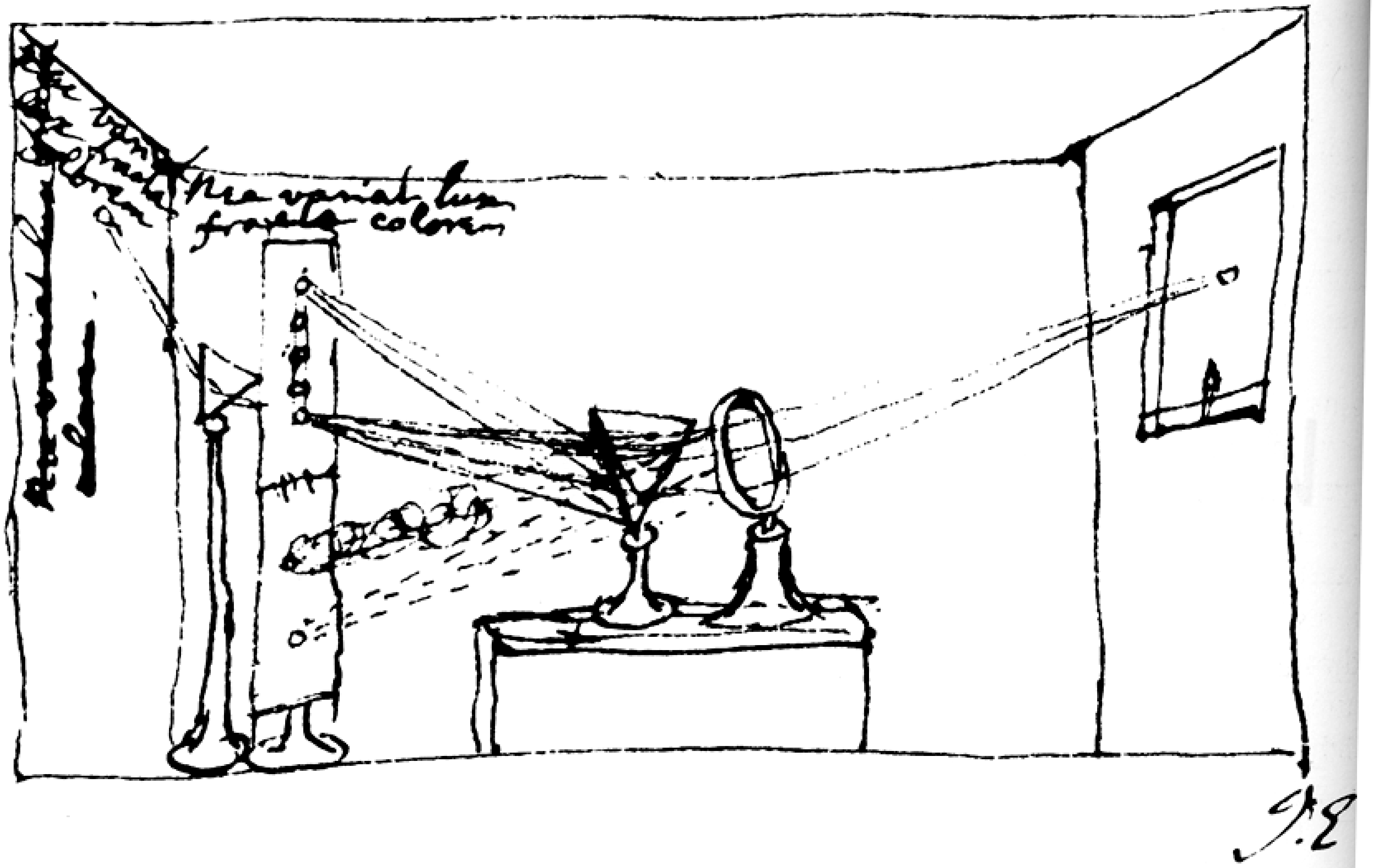
- Review of linear systems, convolution
- Bandpass filter-based image representations
- Probabilistic models for images



Oriented, multi-scale representation

Image

Color

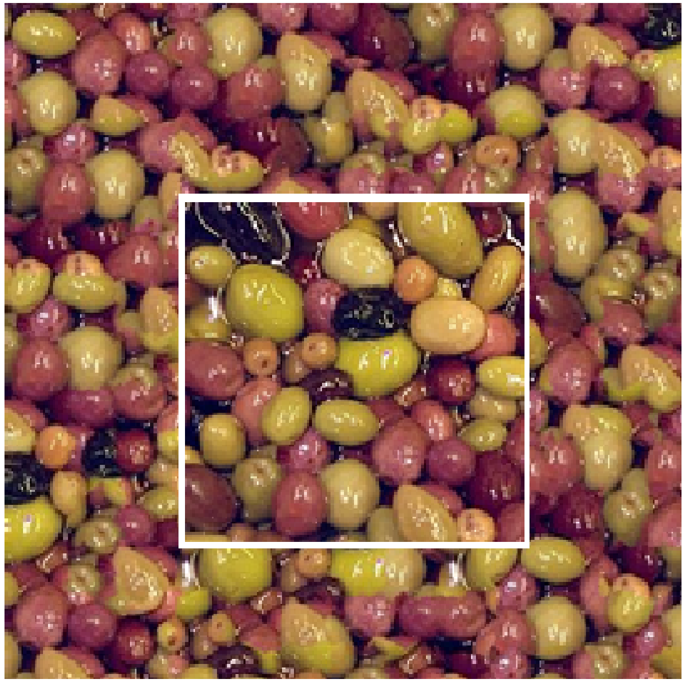


4.1 NEWTON'S SUMMARY DRAWING of his experiments with light. Using a point source of light and a prism, Newton separated sunlight into its fundamental components. By reconverging the rays, he also showed that the decomposition is reversible.

From Foundations of Vision, by Brian Wandell, Sinauer Assoc., 1995

Models of texture

Parametric model



Non-parametric model

A Parametric Texture Model based on Joint Statistics of Complex Wavelet Coefficients
J. Portilla and E. Simoncelli, *International Journal of Computer Vision 40(1): 49-71, October 2000.*
© Kluwer Academic Publishers.

A. Efros and W. T Freeman, **Image quilting for texture synthesis and transfer**, SIGGRAPH 2001

Statistical classifiers

- MIT Media Lab face localization results.

- Applications: database search, human machine interaction, video conferencing.

Multi-view Geometry

What are the relationships between images of point features in more than one view?

Given a point feature in one camera view, predict it’s location in a second (or third) camera?

Ego-Motion / “Match-move”

Where are the cameras?

Track points, estimate consistent poses…

Render synthetic objects in real world!

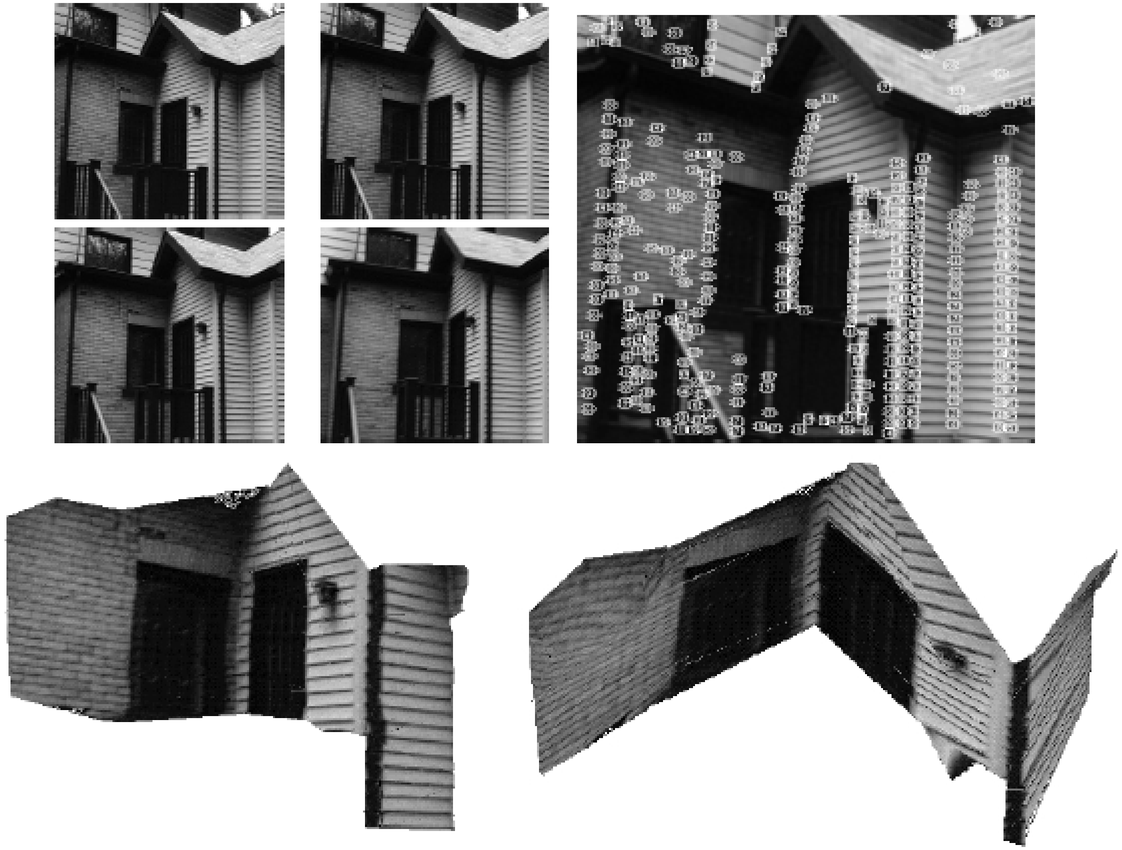
Ego-Motion / “Match-move”

Video

See “Harts War” and other examples in Gallery of examples for Matchmove program at www.realviz.com

Structure from Motion

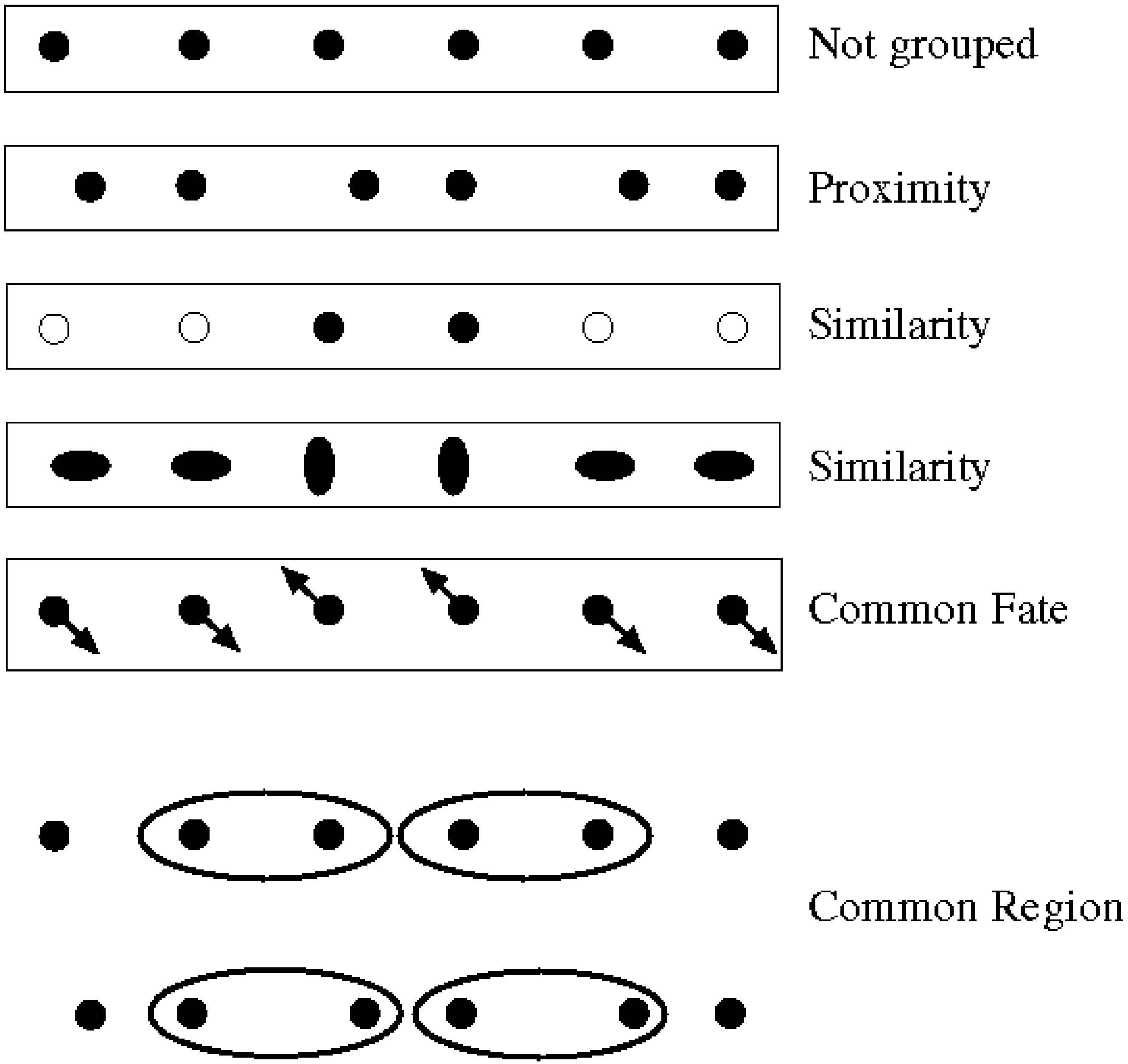
What is the shape of the scene?

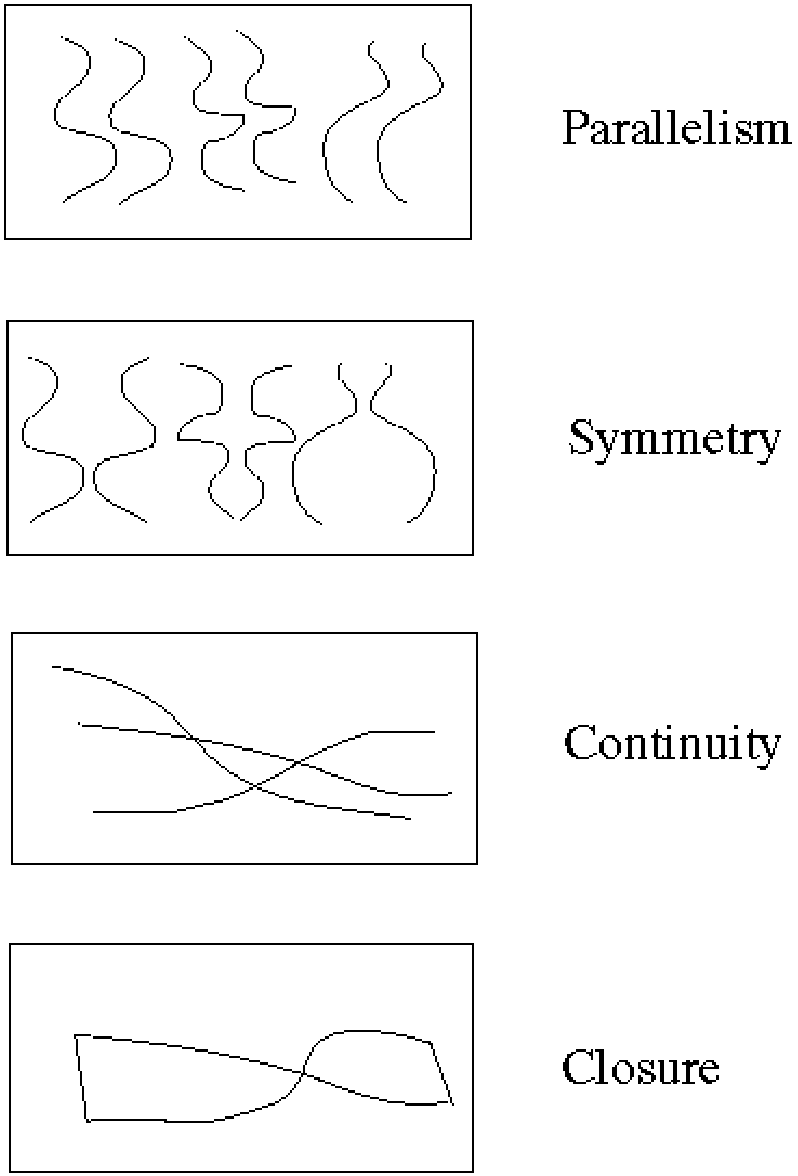


Segmentation

How many ways can you segment six points?

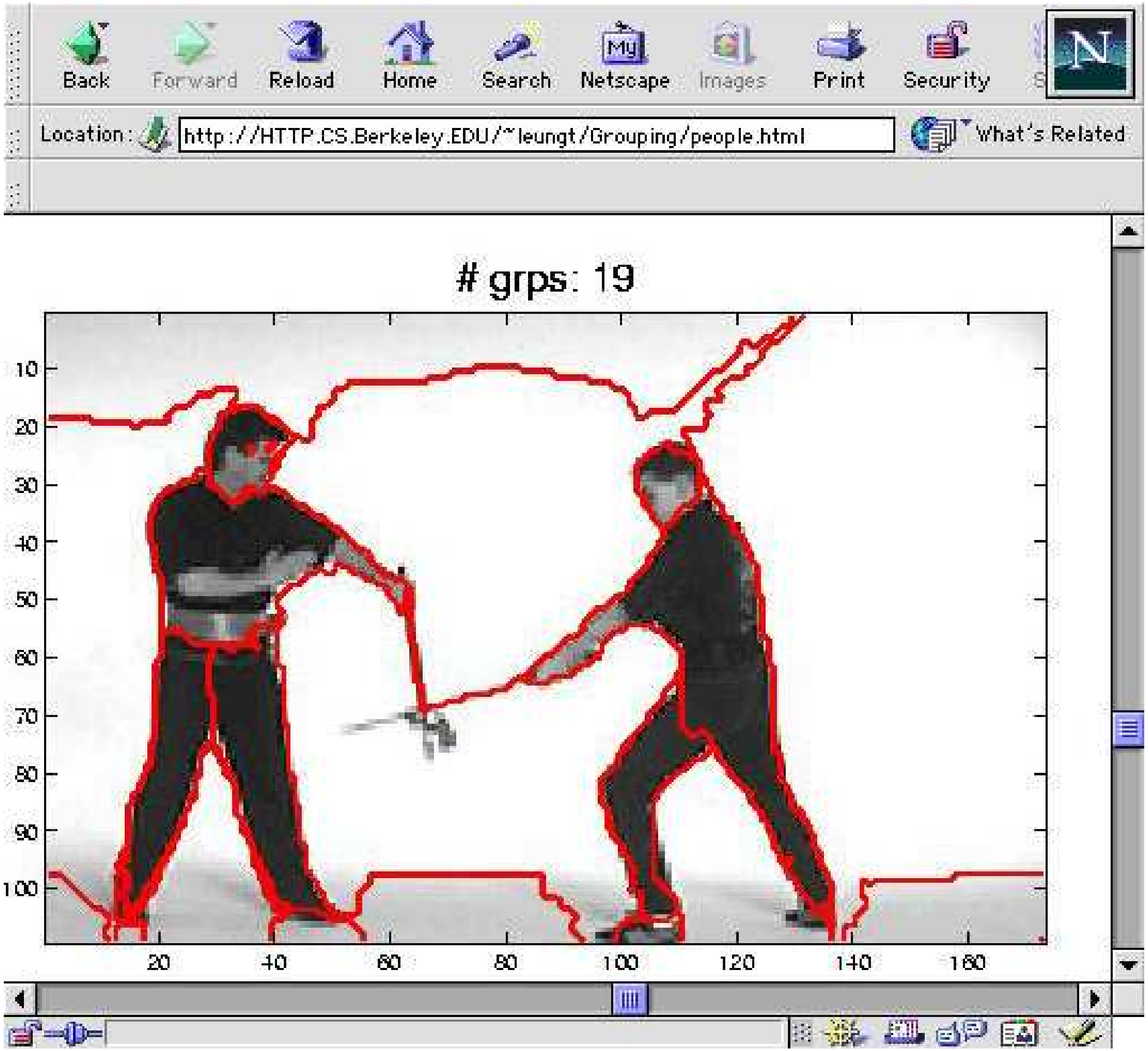
(or curves)

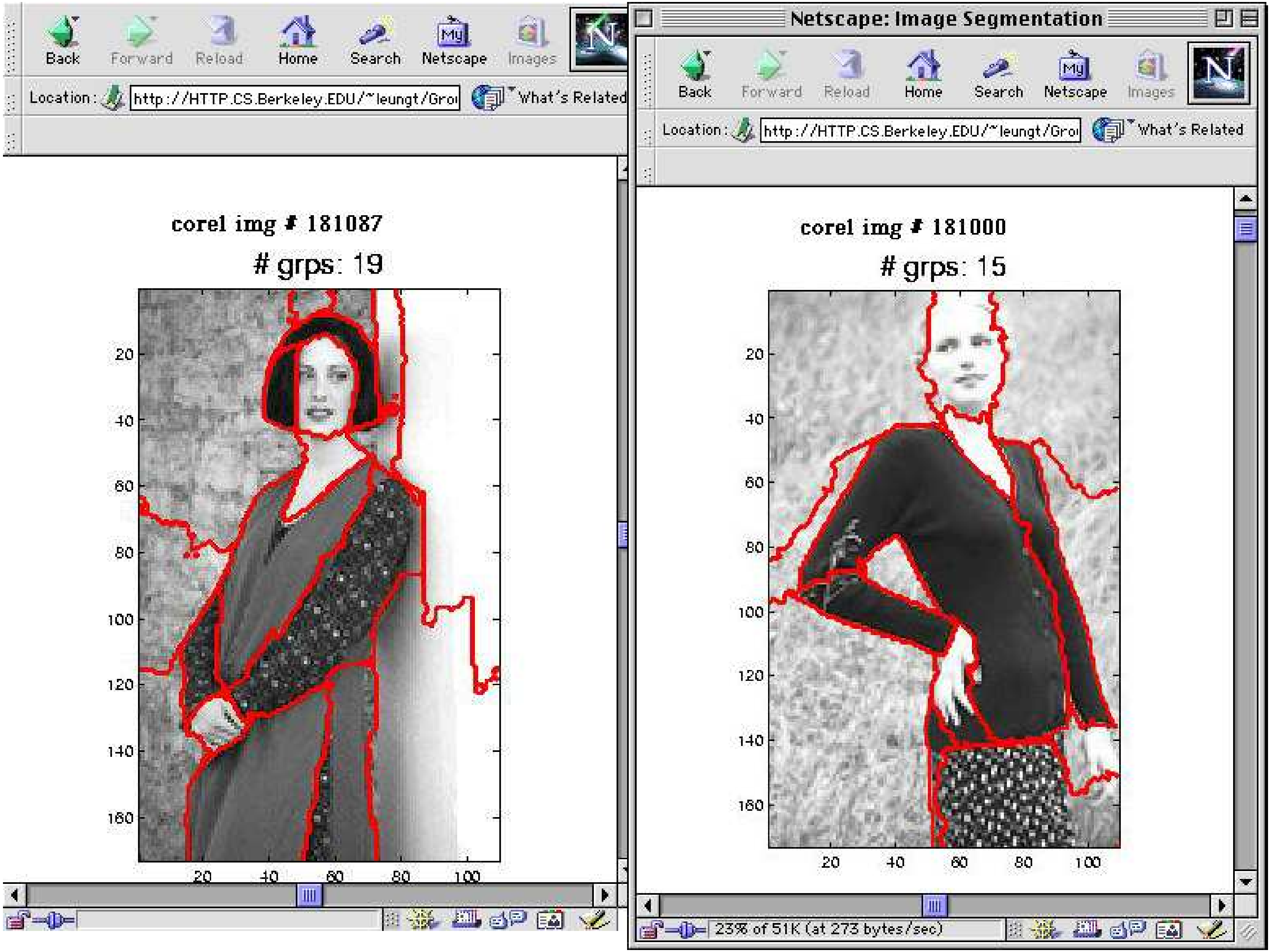

Not grouped
Proximity
Similarity
Similarity
Common Fate
Common Region

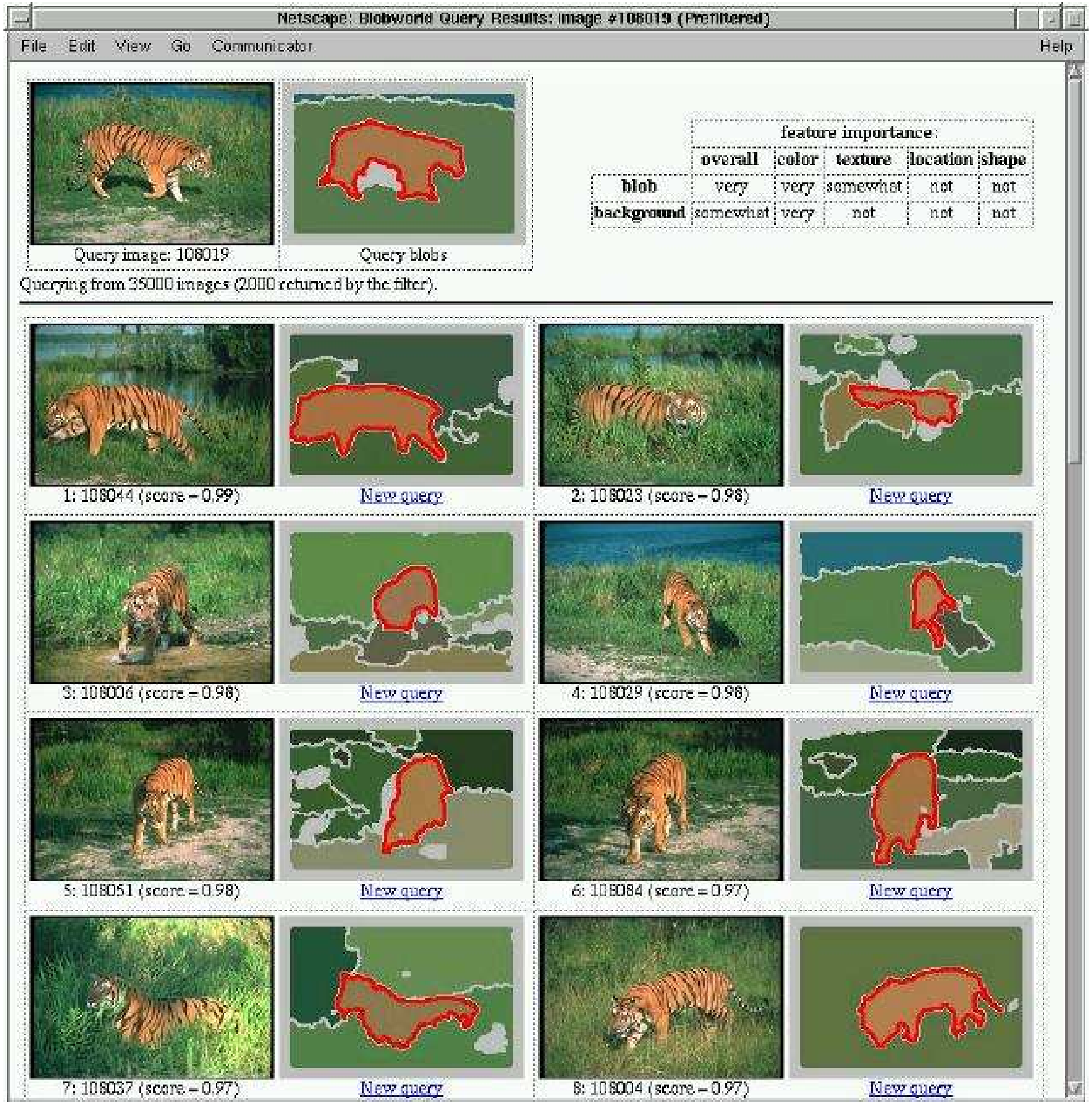

Parallelism
Symmetry
Continuity
Closure

Segmentation

- Which image components “belong together”?
- Belong together=lie on the same object
- Cues
  - similar colour
  - similar texture
  - not separated by contour
  - form a suggestive shape when assembled




Back
Forward
Reload
Home
Search
Netscape
Images
Location: http://HTTP.CS.Berkeley.EDU/~leungt/Gro
What's Related
corel img # 181087
grps: 19
20
40
60
80
100
120
140
160
20
40
60
80
100
Netscape: Image Segmentation
Back
Forward
Reload
Home
Search
Netscape
Images
Location: http://HTTP.CS.Berkeley.EDU/~leungt/Gro
What's Related
corel img # 181000
grps: 15
20
40
60
80
100
120
140
160
20
40
60
80
100
23% of 51K (at 273 bytes/sec)



Query image: 108019

Query blobs

	feature importance:				
	overall	**color**	**texture**	**location**	**shape**
blob	very	very	somewhat	not	not
background	somewhat	very	not	not	not

Querying from 35000 images (2000 returned by the filter).

1: 108044 (score = 0.99)

2: 108023 (score = 0.98)

3: 108006 (score = 0.98)

4: 108029 (score = 0.98)

5: 108051 (score = 0.98)

6: 108084 (score = 0.97)

7: 108037 (score = 0.97)

8: 108004 (score = 0.97)

Tracking

Follow objects and estimate location..

- radar / planes
- pedestrians
- cars
- face features / expressions

Many ad-hoc approaches…

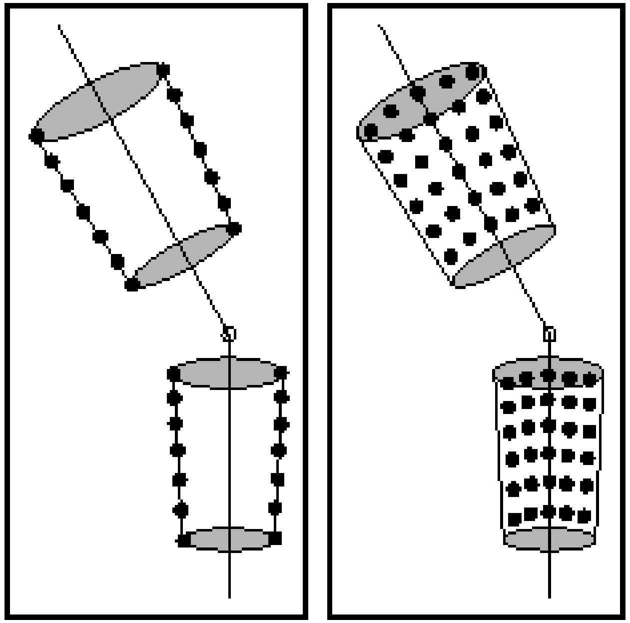
General probabilistic formulation: model density over time.

Tracking

- Use a model to predict next position and refine using next image
- Model:
    - simple dynamic models (second order dynamics)
    - kinematic models
    - etc.
- Face tracking and eye tracking now work rather well

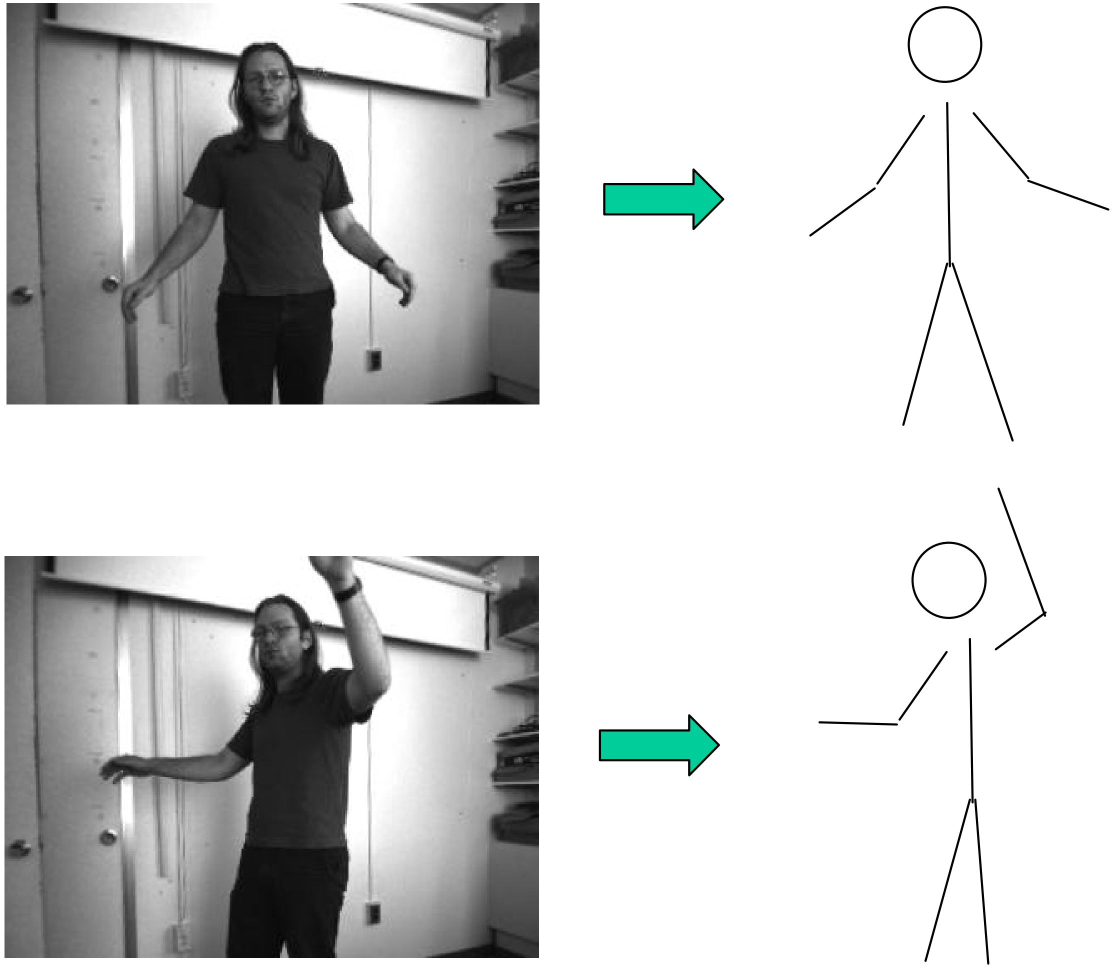
Articulated Models


(a) (b) (c)


(a) (b)

Find most likely model consistent with observations….(and previous configuration)

Articulated tracking



- Constrained optimization
- Coarse-to-fine part iteration
- Propagate joint constraints through each limb
- Real-time on Ghz pentium…


slow

And…

- Visual Category Learning
- Image Databases
- Image-based Rendering
- Visual Speechreading
- Medical Imaging

Administrivia

- Syllabus
- Grading
- Collaboration Policy
- Project

Lecture	Date	Description	Readings	Assignments	Mater
1	2/3	Course Introduction Cameras, Lenses and Sensors	Req: FP 1.1, 2.1, 2.2, 2.3, 3.1, 3.2	PS0 out	
2	2/5	Image Filtering	Req: FP 7.1 - 7.6		
3	2/10	Image Representations: pyramids	Req: FP 7.7, 9.2		
4	2/12	Texture	Req: FP 9.1, 9.3, 9.4	PS0 due	
	2/17	Monday Classes Held (NO LECTURE)			
5	2/19	Color	Req: FP 6.1-6.4	PS1 out	
6	2/24	Local Features			
7	2/26	Multiview Geometry	Req: FP 10	PS1 due	
8	3/2	Affine Reconstruction	Req: FP 12		
9	3/4	Projective Reconstruction	Req: FP 13	PS2 out	
10	3/9	Scene Reconstruction			
11	3/11	Non-Rigid Motion		PS2 due	
12	3/16	Morphable and Active Appearance Models		EX1 out	
13	3/18	Model-Based Object Recognition		EX1 due	
	3/23-3/25	Spring Break (NO LECTURE)			

	3/23-3/25	Spring Break (NO LECTURE)			
14	3/30	Face Detection and Recognition I			
15	4/1	Face Detection and Recognition II		Project proposal due	
16	4/6	Category Learning			
17	4/9	Segmentation I		PS3 out	
18	4/13	Segmentation II			
19	4/15	Medical Imaging		PS3 due	
20	4/20	Tracking I			
21	4/22	Tracking II		PS4 out	
22	4/27	Image-Based Rendering			
23	4/29	Example-based inference		PS4 due	
24	5/4	Multimodal Interfaces		EX2 out	
25	5/6	Image Databases		EX2 due	
26	5/11	Project Presentations 11-2pm			
27	5/13	Projects Due--no class		Project final report due (extension to 5/16 on request)	

Grading

- Two take-home exams
- Five problem sets with lab exercises in Matlab
- No final exam
- Final project

Collaboration Policy

Problem sets may be discussed, but all written work and coding must be done individually. Take-home exams may not be discussed. Individuals found submitting duplicate or substantially similar materials due to inappropriate collaboration may get an F in this class and other sanctions.

Project

The final project may be

- An original implementation of a new or published idea
- A detailed empirical evaluation of an existing implementation of one or more methods
- A paper comparing three or more papers not covered in class, or surveying recent literature in a particular area

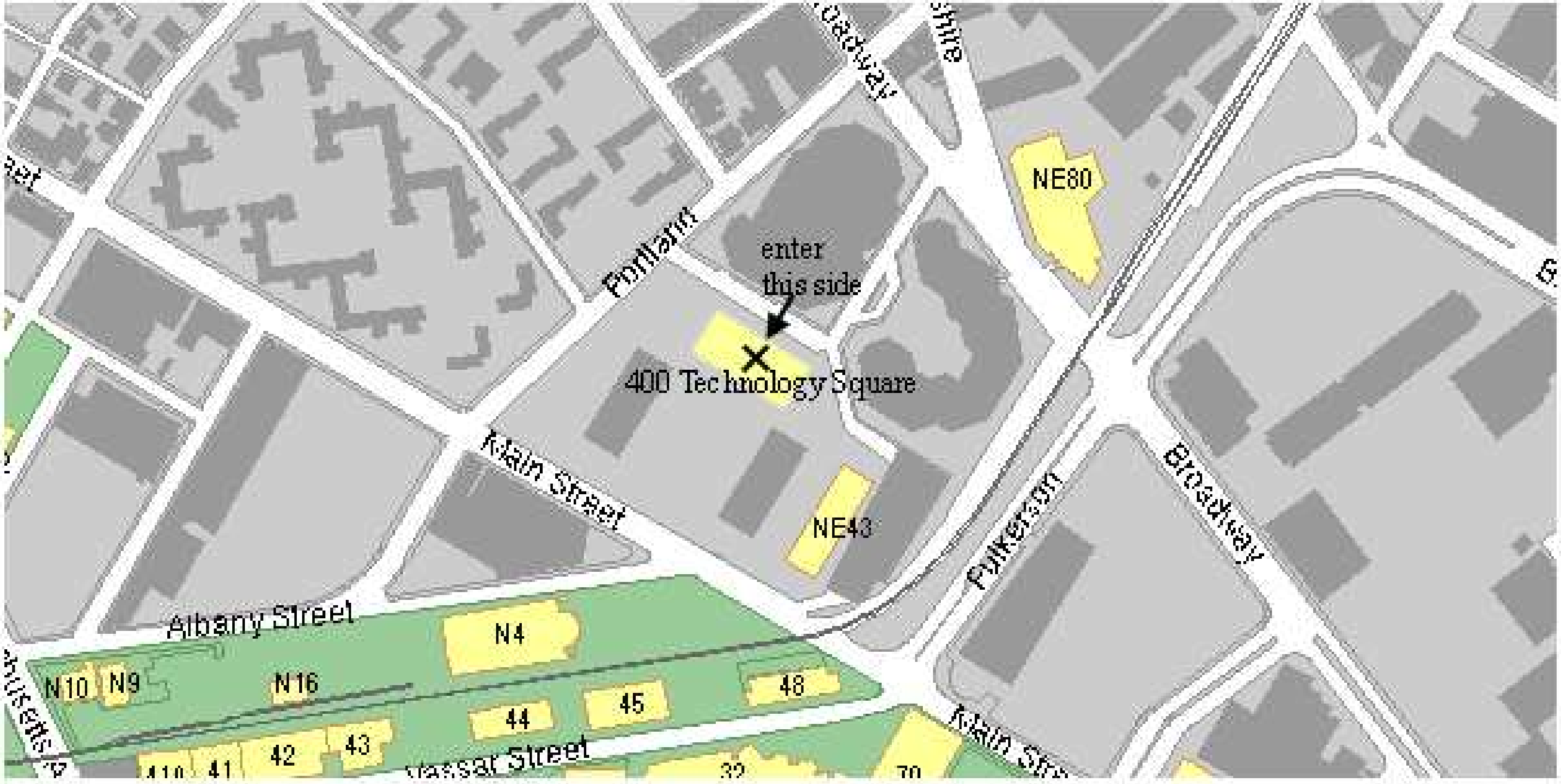
A project proposal not longer than two pages must be submitted and approved by April 1st.

Problem Set 0

- Out today, due 2/12
- Matlab image exercises
  - load, display images
  - pixel manipulation
  - RGB color interpolation
  - image warping / morphing with `interp2`
  - simple background subtraction
- All psets graded loosely: *check, check-, 0.*
- (Outstanding solutions get extra credit.)

Map showing 400 Technology Square

The building says "Forrester" on the side. Only the parking garage side building entrance is unlocked. (After normal business hours, the elevator to our floor and the building itself are both locked.) Exiting the elevator on the 6th floor, you'll see a pair of glass doors on one side. Enter the left glass door, then turn right at every opportunity to find my office, room 601.



back to my home page , Sept., 2002.

Cameras, lenses, and calibration

Today:

- Camera models (review)
- Projection equations (review)

You should have been exposed to this material in previous courses; this lecture is just a (quick) review.

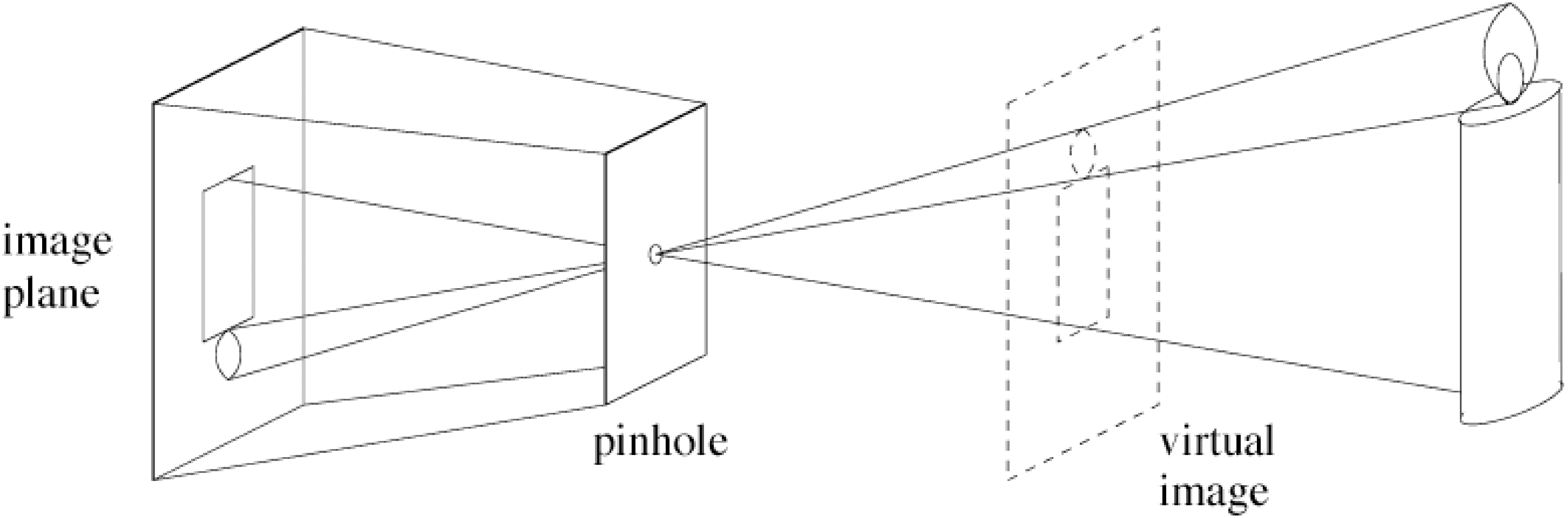
- Calibration methods (new)

7-year old's question

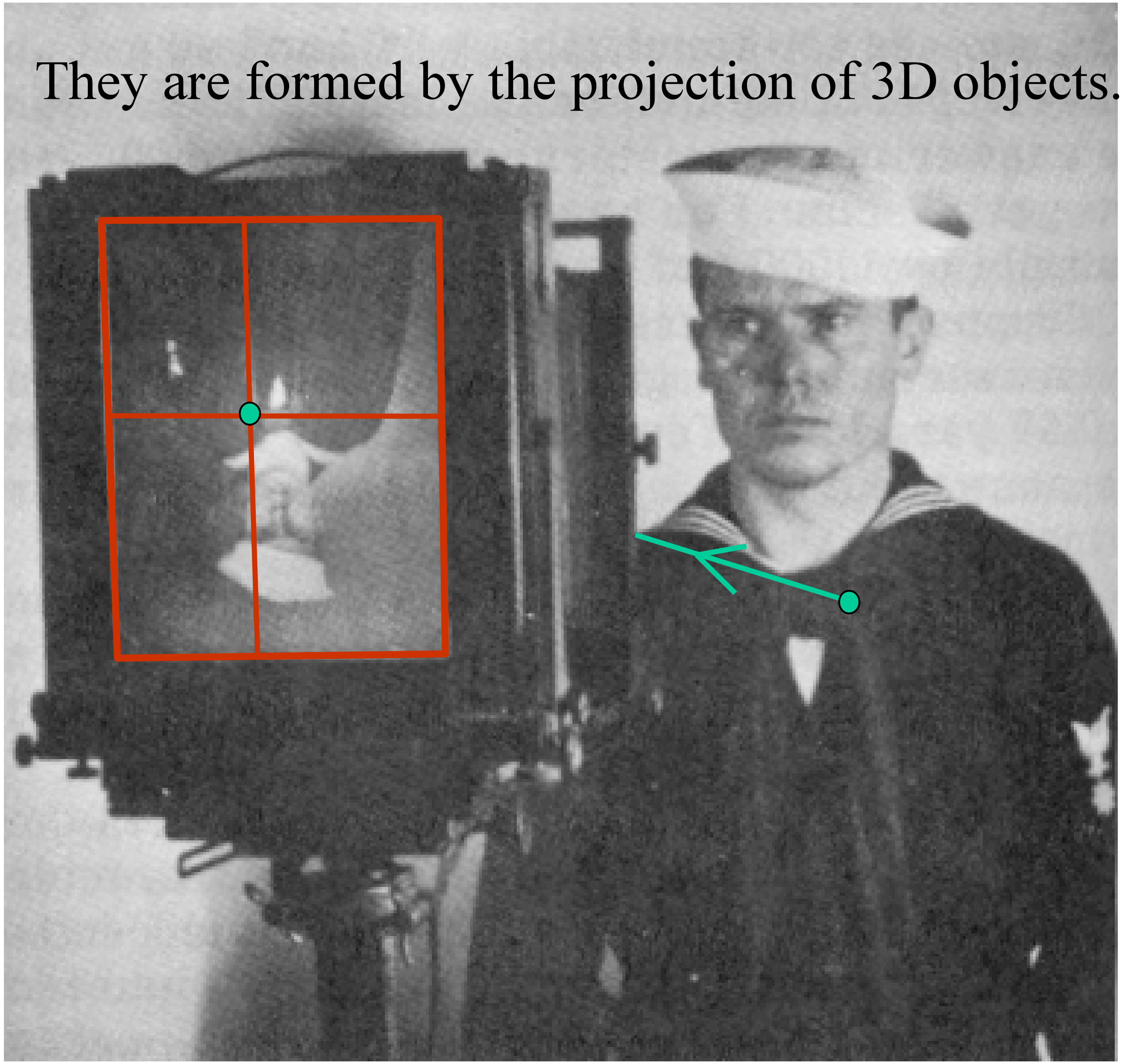
Why is there no image on a white piece of paper?

Virtual image, perspective projection

- Abstract camera model - box with a small hole in it



Forsyth&Ponce

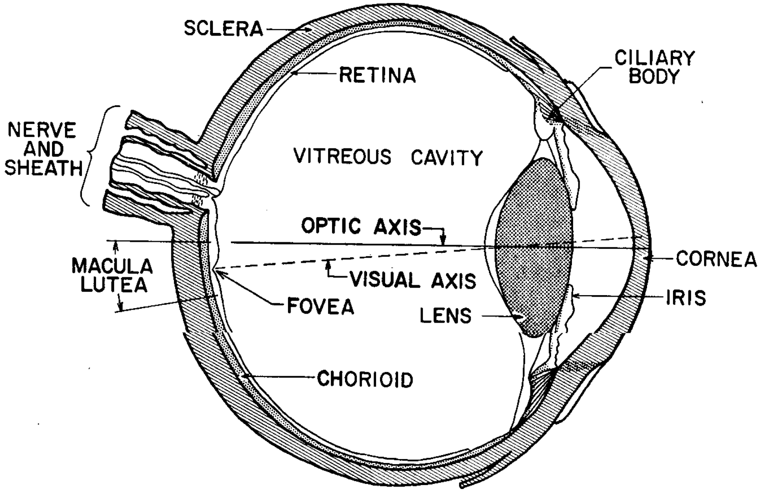


They are formed by the projection of 3D objects.

Figure from US Navy Manual of Basic Optics and Optical Instruments, prepared by Bureau of Naval Personnel. Reprinted by Dover Publications, Inc., 1969.

Images are two-dimensional patterns of brightness values.

Reproduced by permission, the American Society of Photogrammetry and Remote Sensing. A.L. Nowicki, “Stereoscopy.” Manual of Photogrammetry, Thompson, Radlinski, and Speert (eds.), third edition, 1966.



Animal eye: a looonnng time ago.

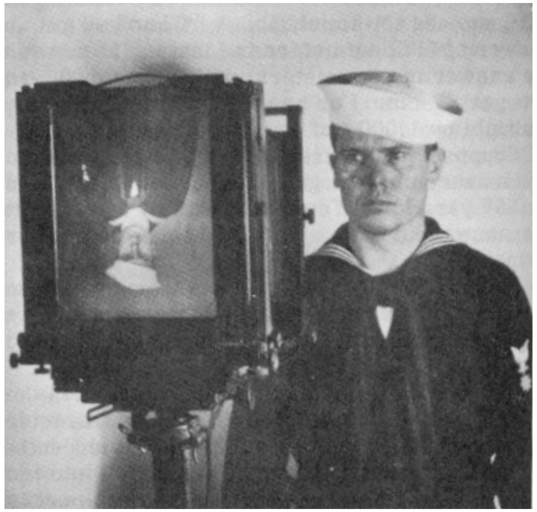
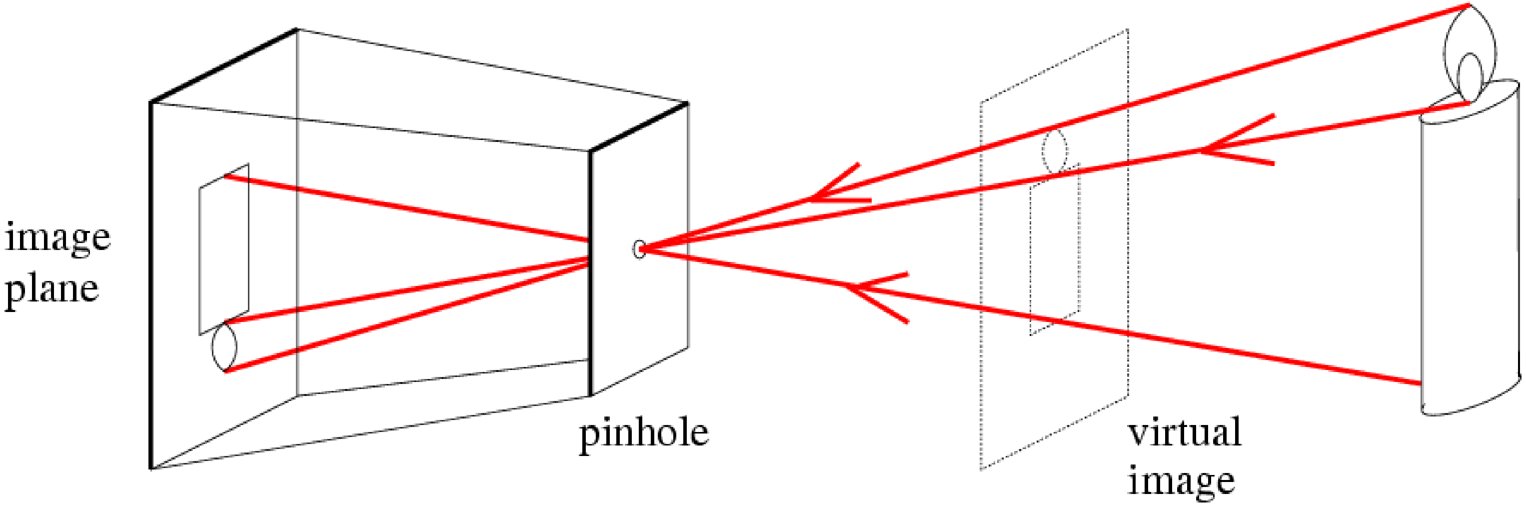


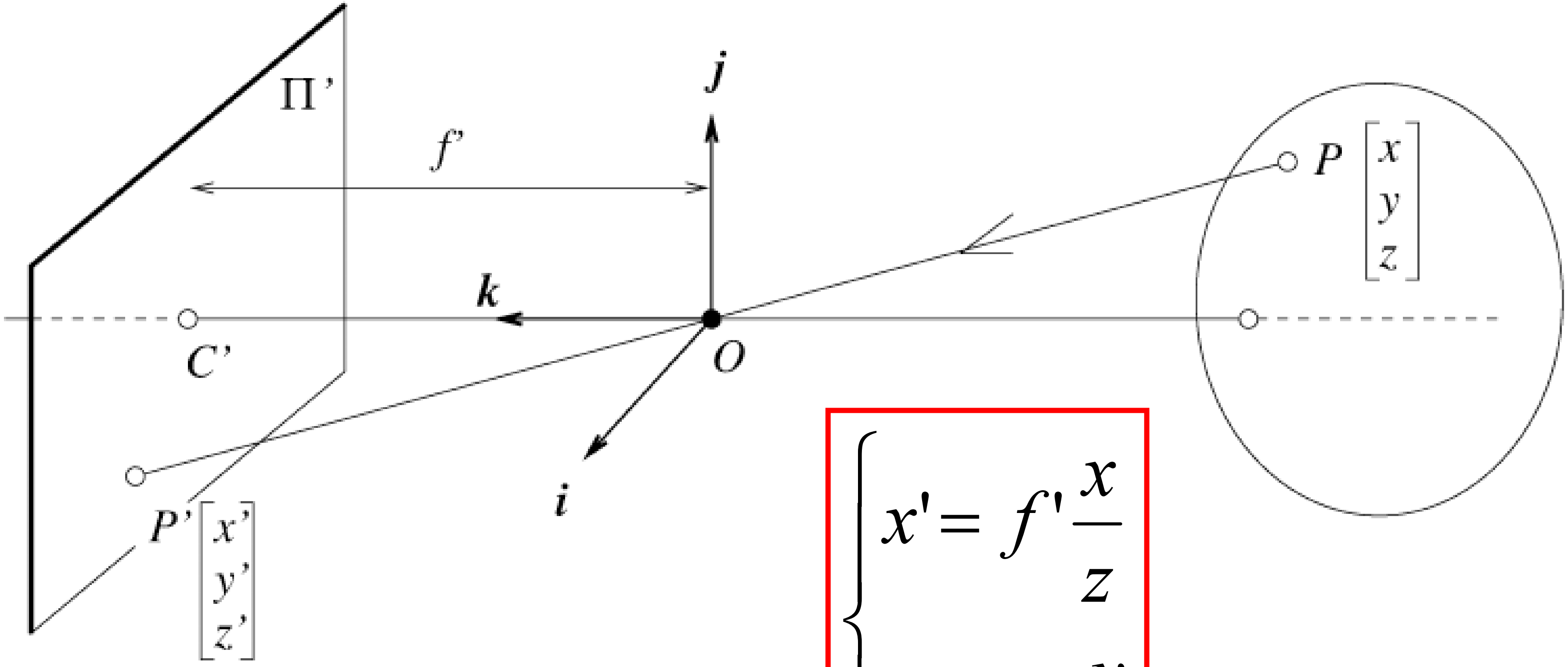
Figure from US Navy Manual of Basic Optics and Optical Instruments, prepared by Bureau of Naval Personnel. Reprinted by Dover Publications, Inc., 1969.

Photographic camera: Niepce, 1816.



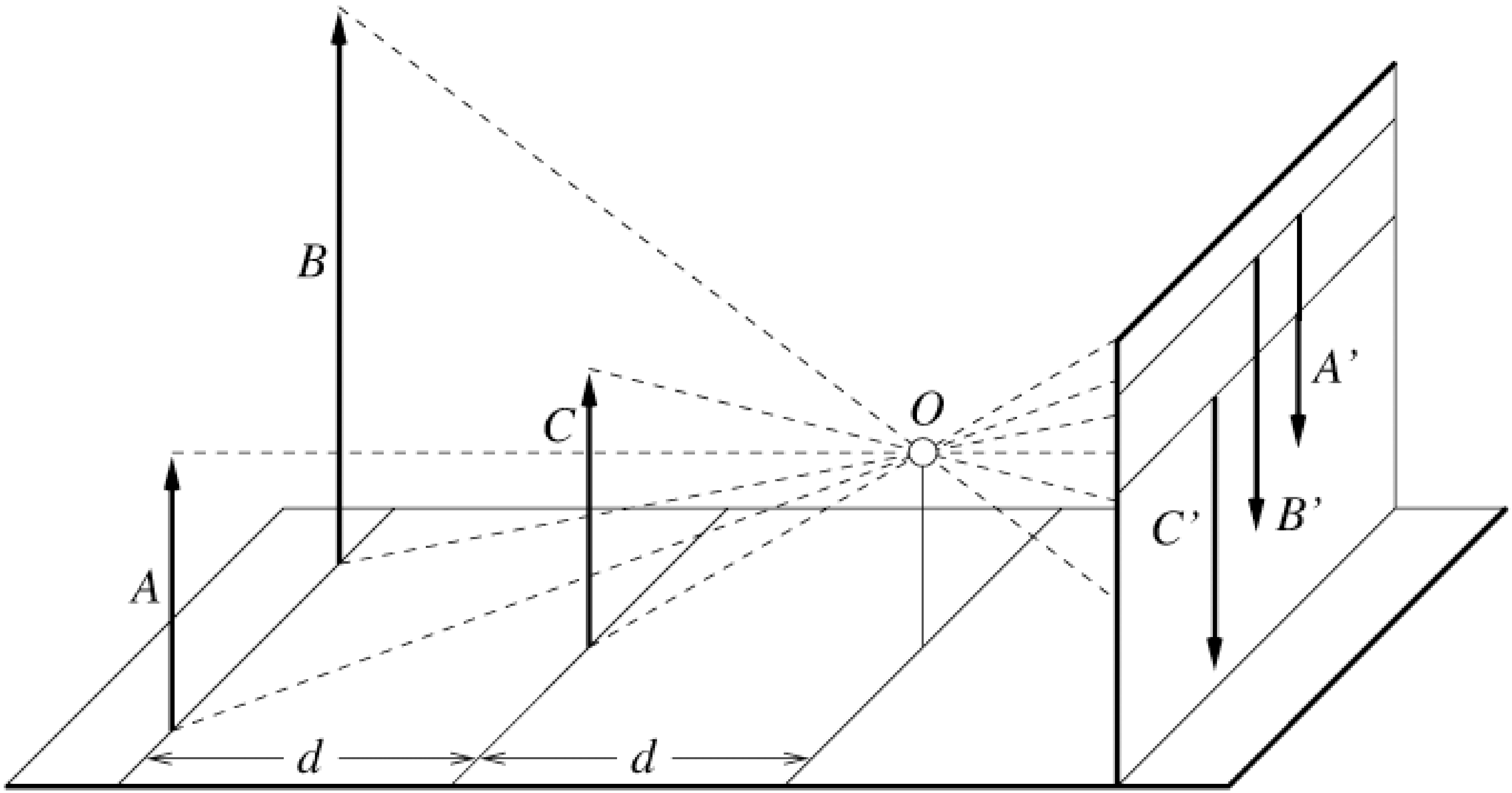
Pinhole perspective projection: Brunelleschi, XVth Century.
Camera obscura: XVIth Century.

The equation of projection



$$\begin{cases} x' = f'\dfrac{x}{z} \\[2ex] y' = f'\dfrac{y}{z} \end{cases}$$

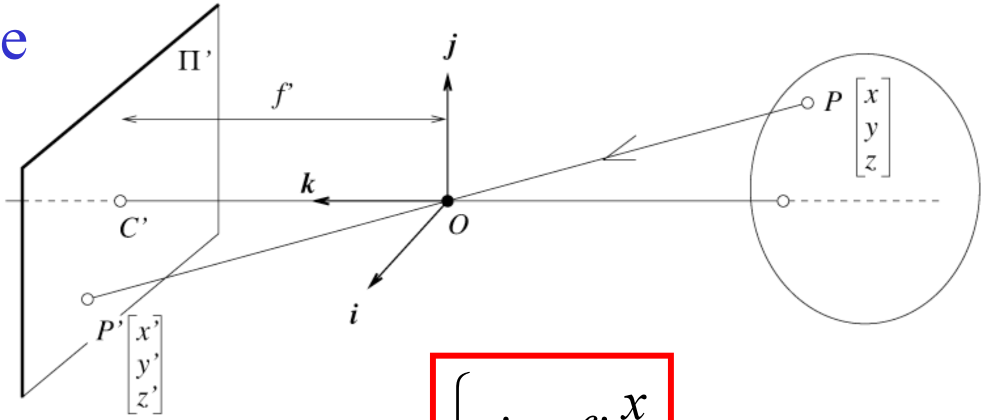
Distant objects are smaller



Forsyth&Ponce

Geometric properties of projection

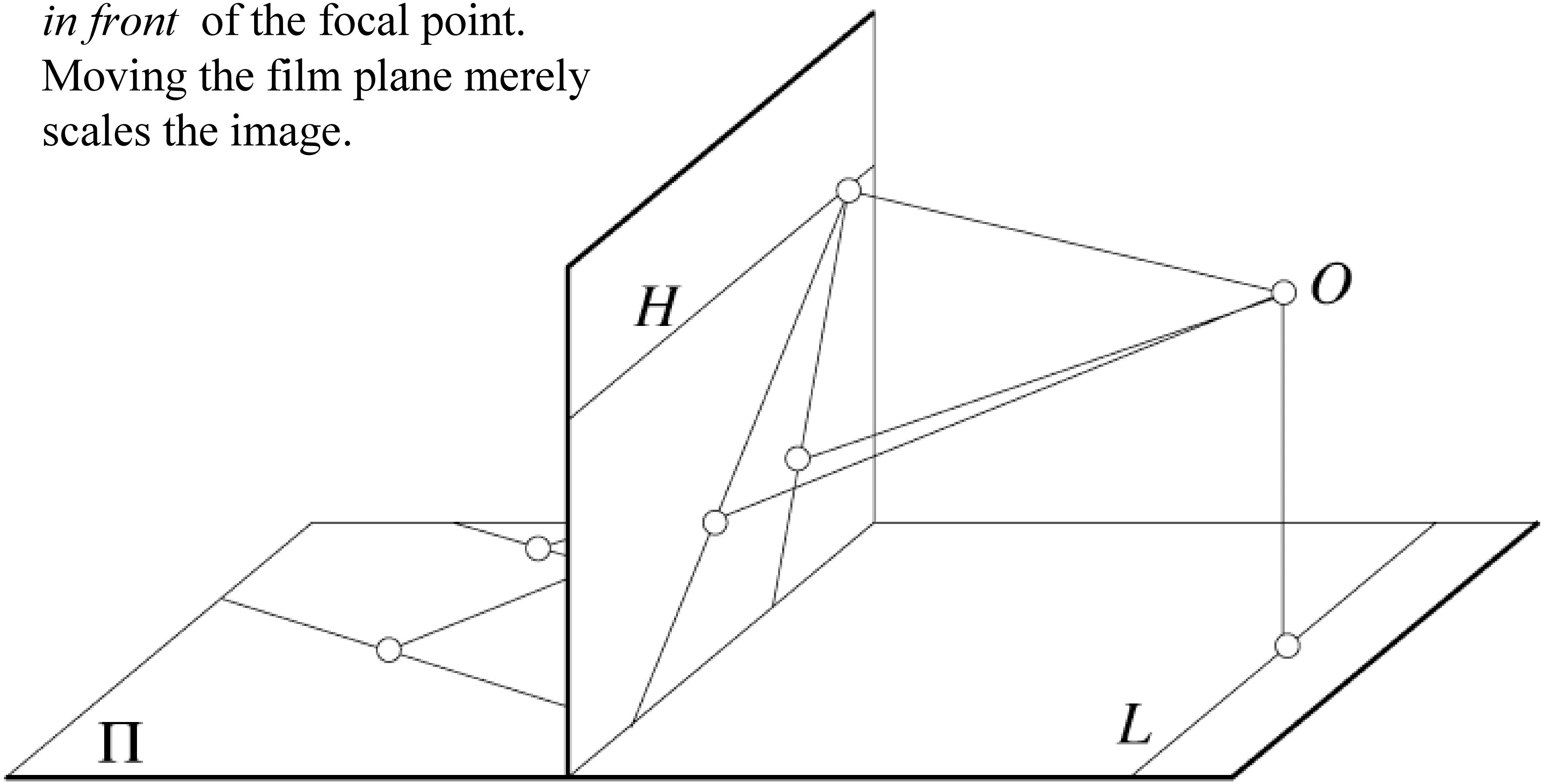
- Points go to points
- Lines go to lines
- Planes go to the whole image or a half-plane
- Polygons go to polygons
- Degenerate cases
  - line through focal point to point
  - plane through focal point to line



$$\begin{cases} x' = f'\dfrac{x}{z} \\ y' = f'\dfrac{y}{z} \end{cases}$$

Parallel lines meet

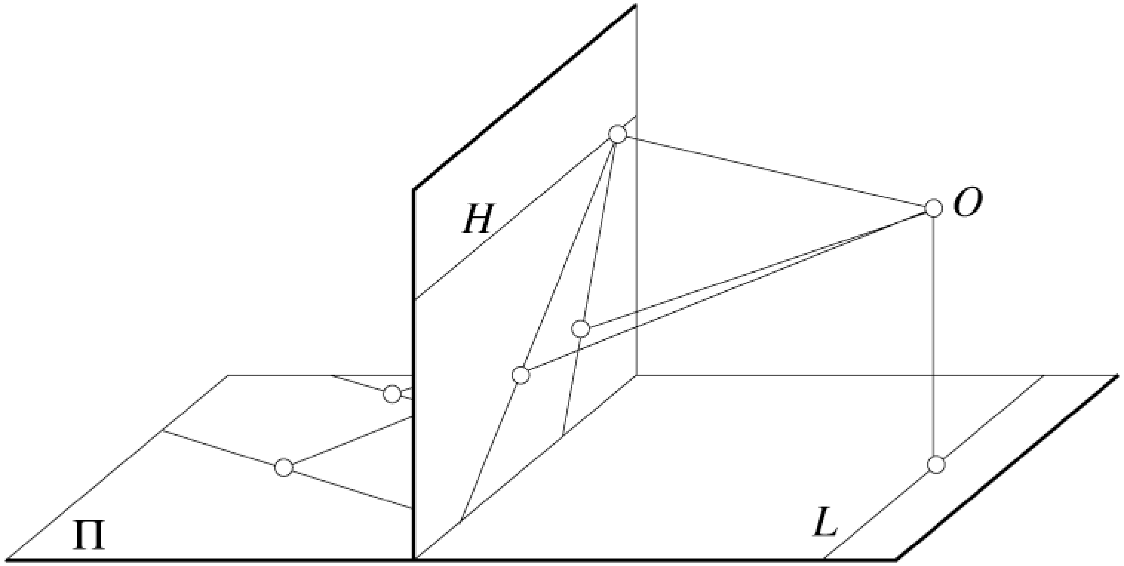
Common to draw film plane
in front of the focal point.
Moving the film plane merely
scales the image.



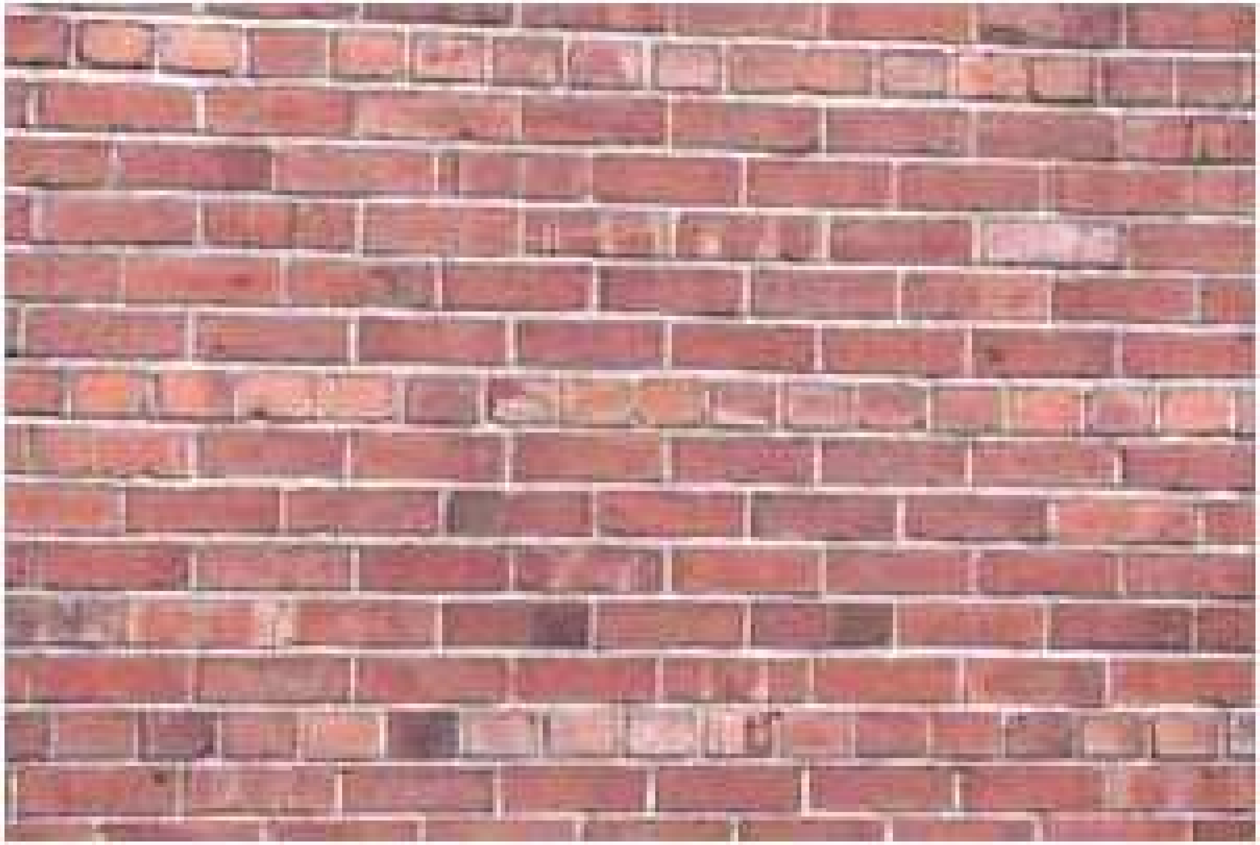
Forsyth&Ponce

Vanishing points

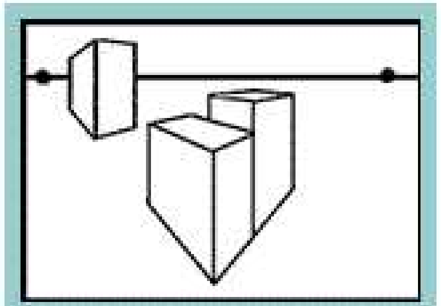
- Each set of parallel lines (=direction) meets at a different point
  - The *vanishing point* for this direction
- Sets of parallel lines on the same plane lead to *collinear* vanishing points.
  - The line is called the *horizon* for that plane



What if you photograph a brick wall head-on?



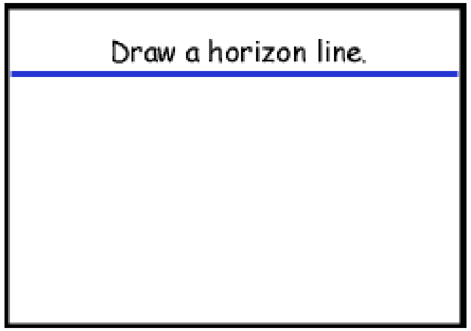
Two-point perspective

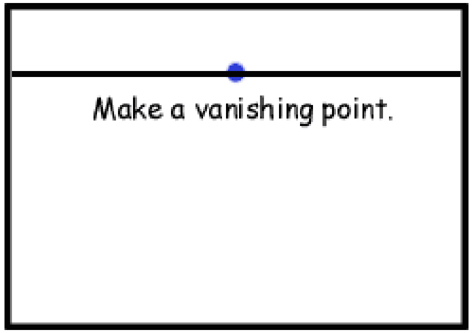


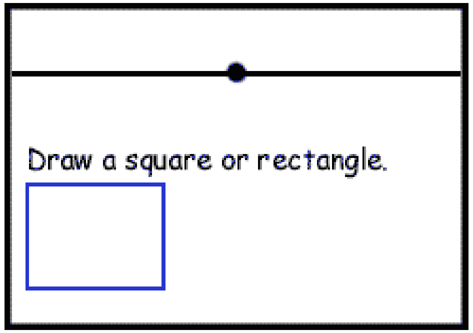
It's easy to draw simple forms in two-point perspective.

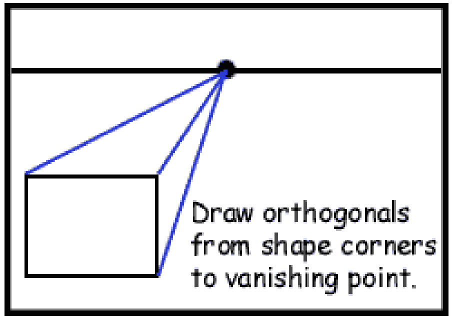
Linear perspective allows artists to trick the eye into seeing depth on a flat surface.

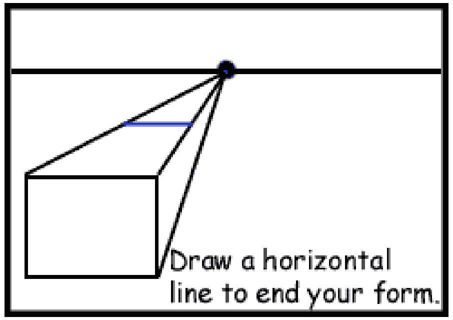
http://www.sanford-artedventures.com/create/tech_2pt_perspective.html

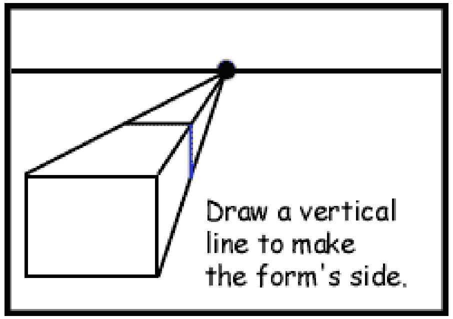

Draw a horizon line.

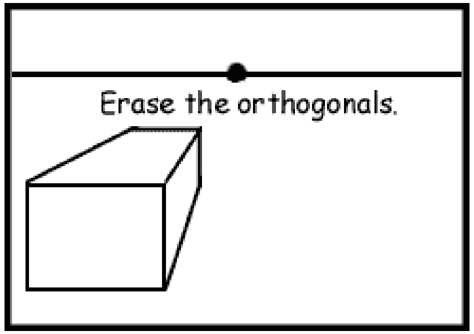

Make a vanishing point.


Draw a square or rectangle.


Draw orthogonals from shape corners to vanishing point.

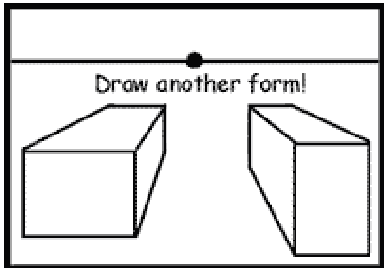

Draw a horizontal line to end your form.

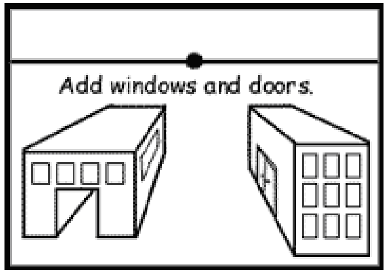

Draw a vertical line to make the form's side.

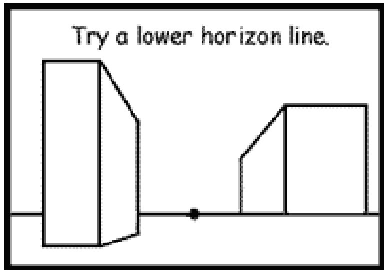

Erase the orthogonals.

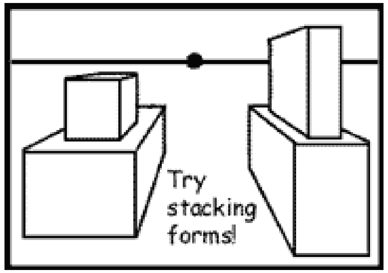
Now you have a 3-D form in one-point perspective!

10. Add details and experiment!

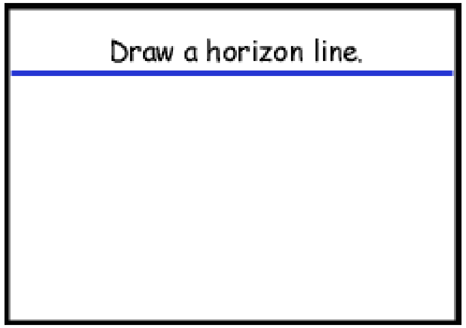

Draw another form!


Add windows and doors.

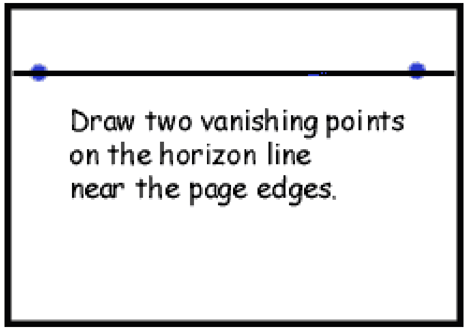

Try a lower horizon line.


Try stacking forms!

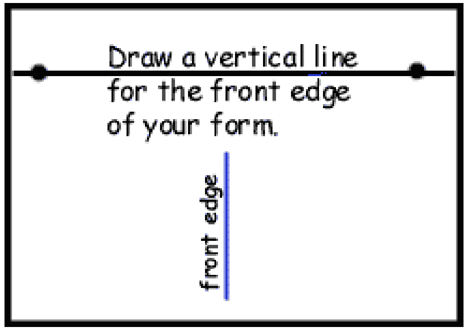
http://www.sanford-artedventures.com/create/tech_1pt_perspective.html



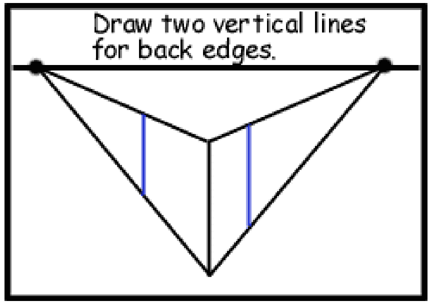
Draw a horizon line.



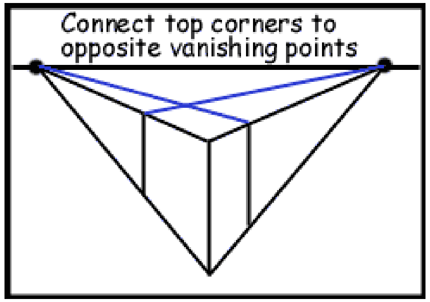
Draw two vanishing points
on the horizon line
near the page edges.



Draw a vertical line
for the front edge
of your form.

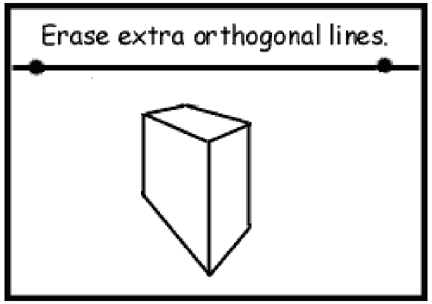


Draw two vertical lines
for back edges.

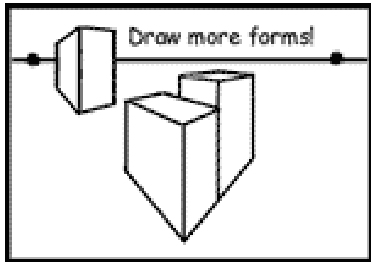


Connect top corners to
opposite vanishing points

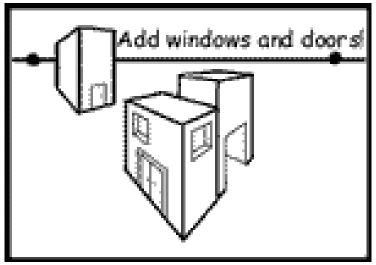
Draw lightly so you can erase!



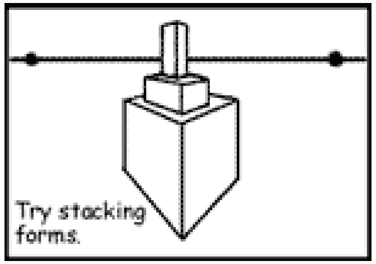
Erase extra orthogonal lines.



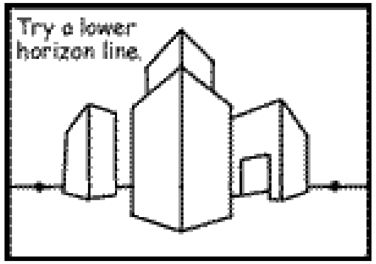
Draw more forms!



Add windows and doors!

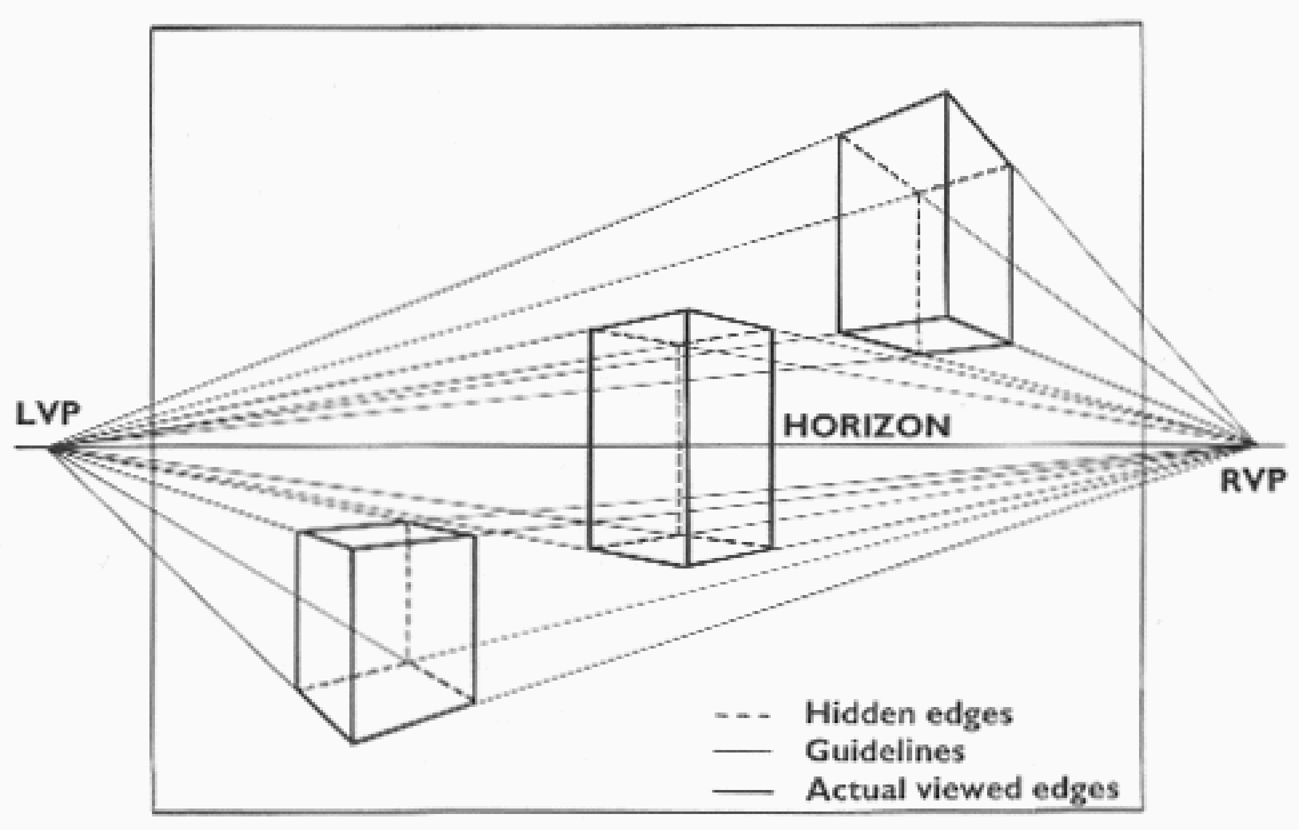


Try stacking forms.



Try a lower horizon line.

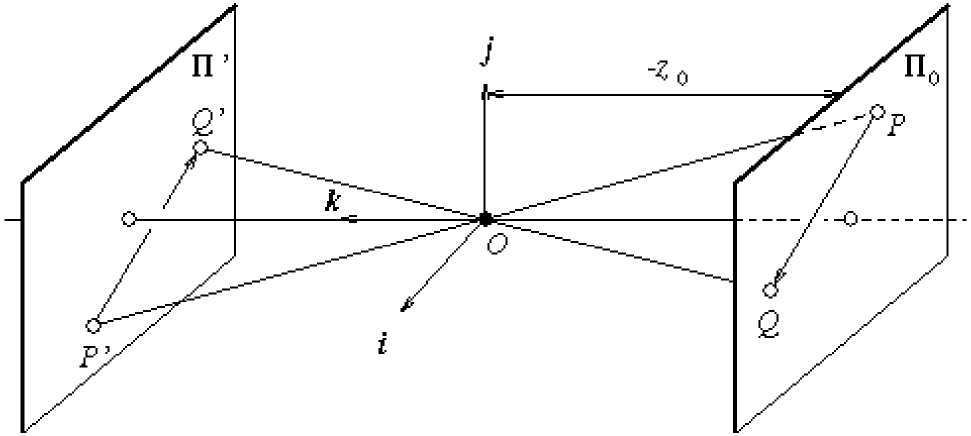
http://www.sanford-artedventures.com/create/tech_2pt_perspective.html



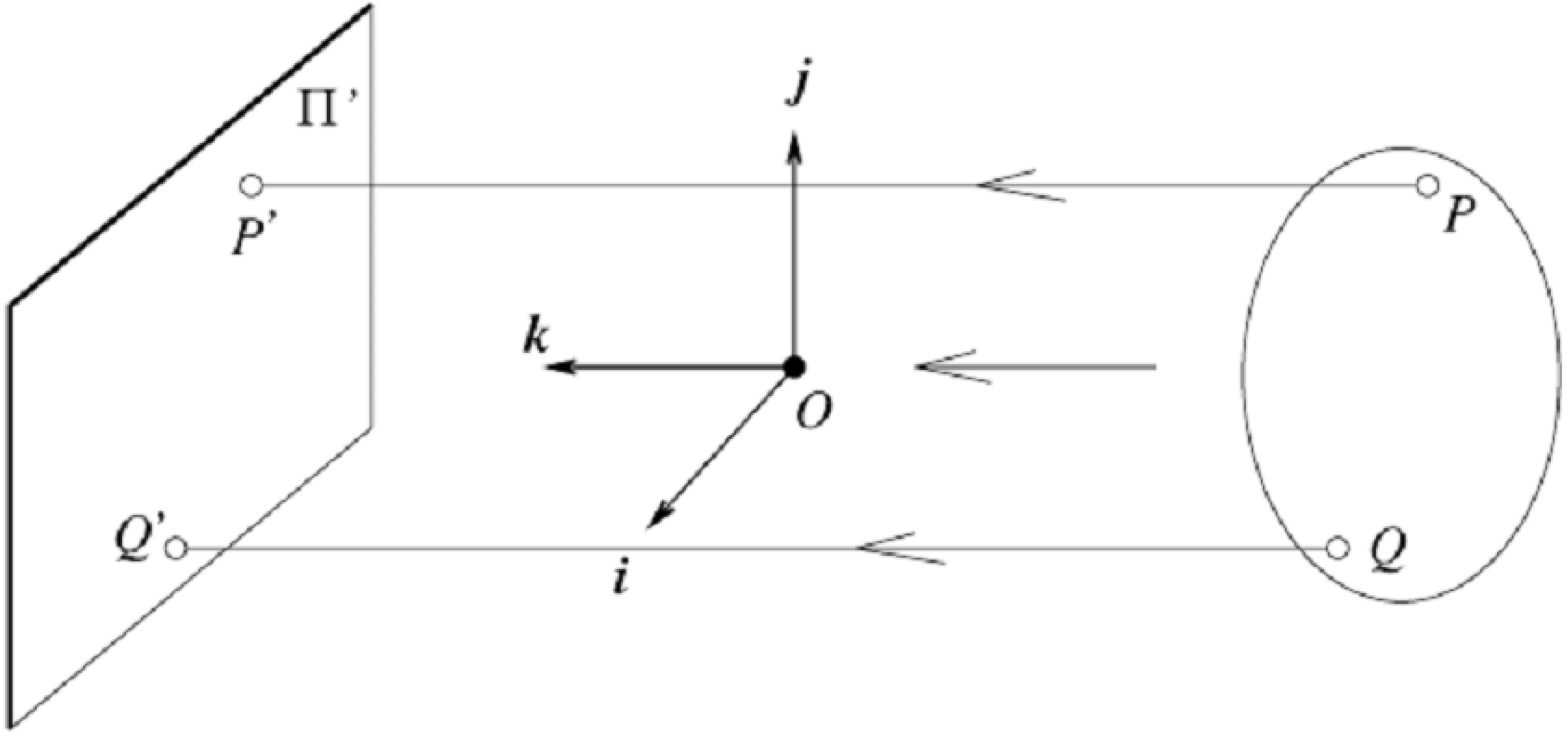
http://www.siggraph.org/education/materials/HyperGraph/viewing/view3d/perspect.htm

Weak perspective

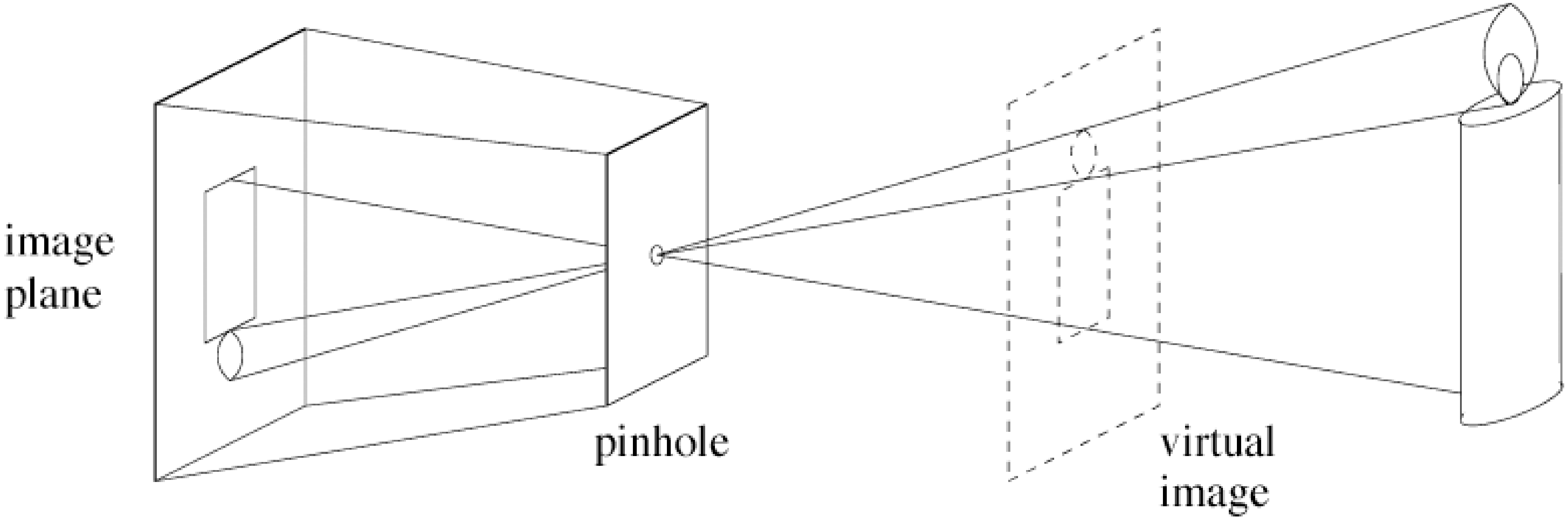
- Issue
  - perspective effects, but not over the scale of individual objects
  - collect points into a group at about the same depth, then divide each point by the depth of its group
  - Adv: easy
  - Disadv: wrong

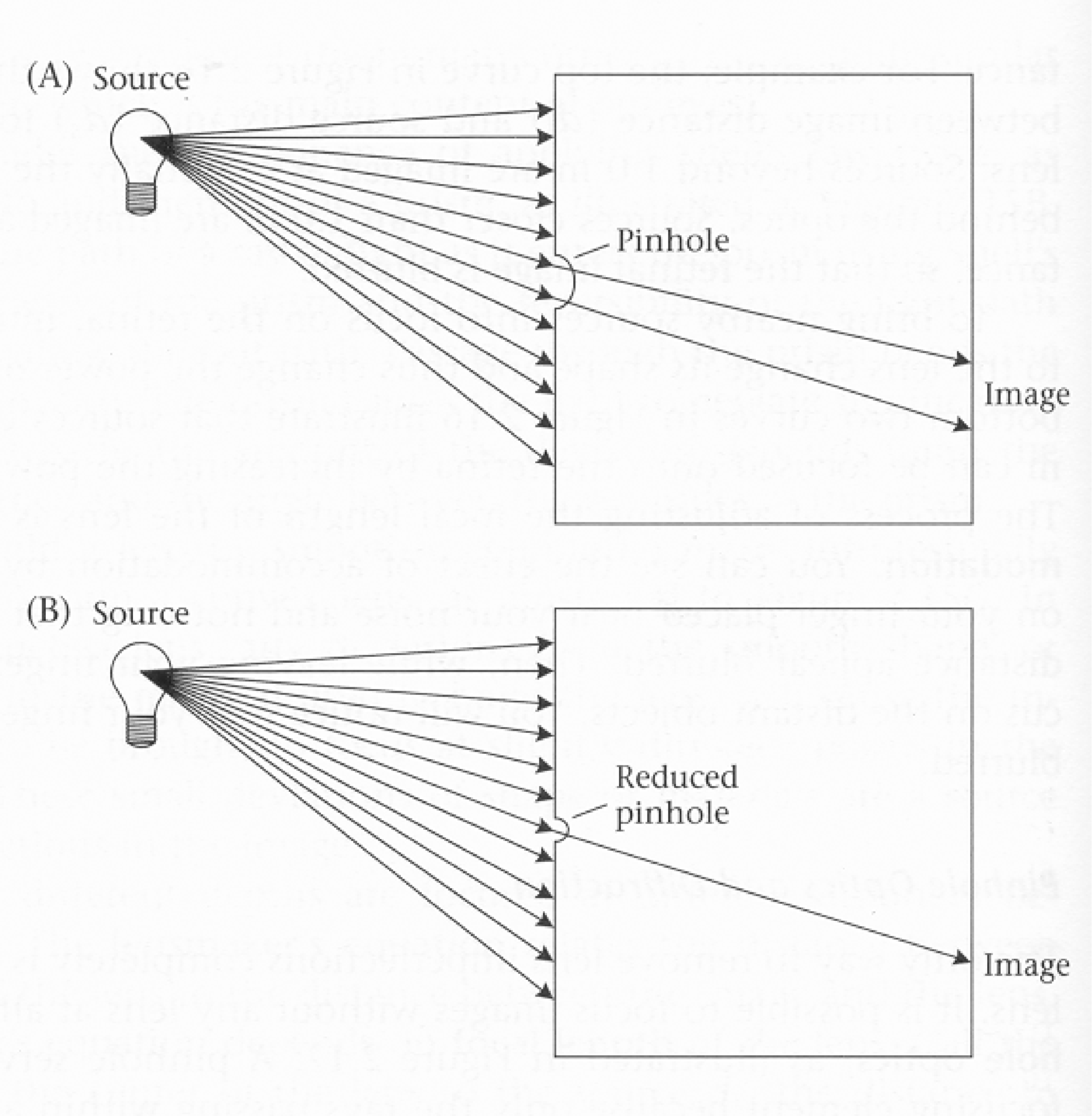


Orthographic projection

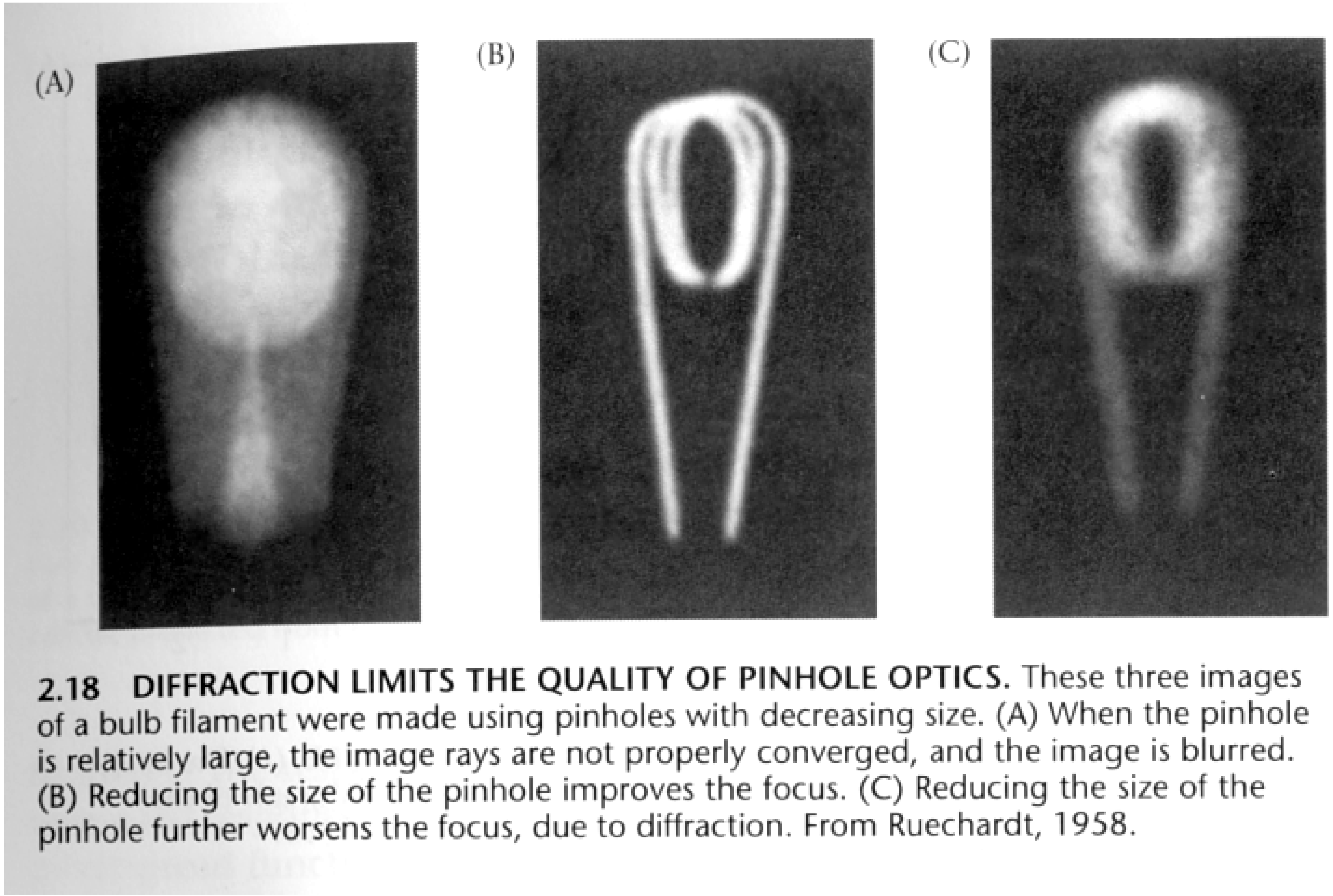


How large a pinhole?





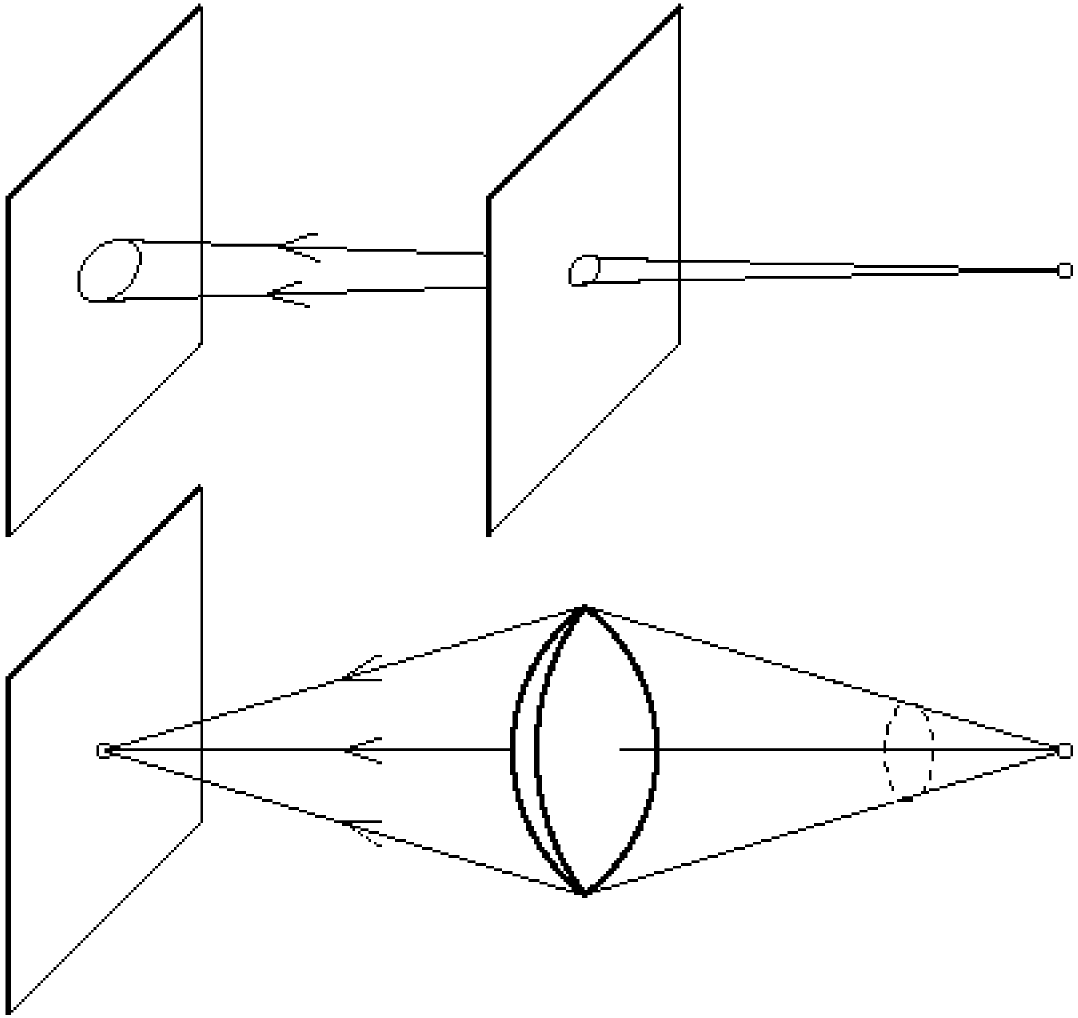
Wandell, Foundations of Vision, Sinauer, 1995



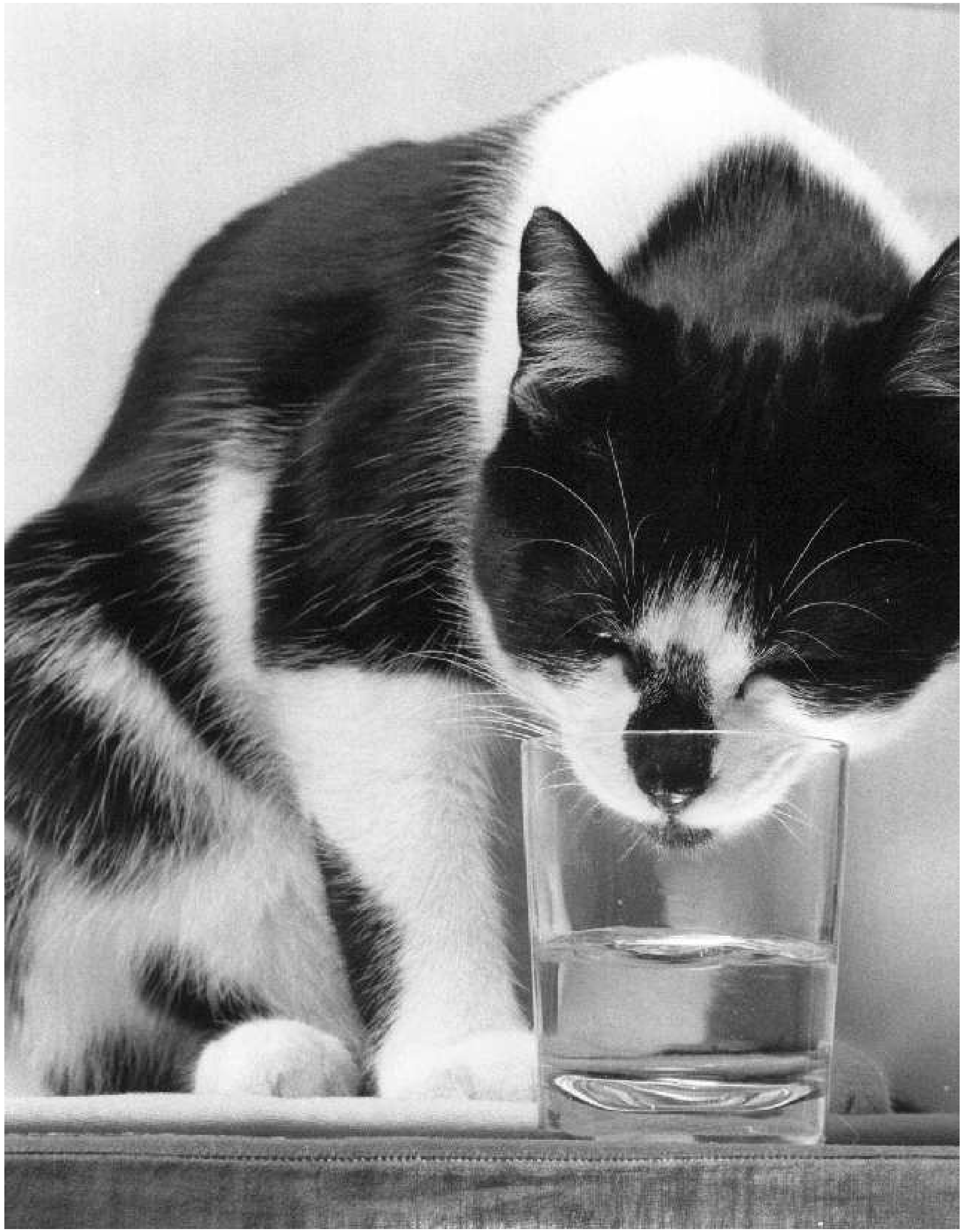
2.18 DIFFRACTION LIMITS THE QUALITY OF PINHOLE OPTICS. These three images of a bulb filament were made using pinholes with decreasing size. (A) When the pinhole is relatively large, the image rays are not properly converged, and the image is blurred. (B) Reducing the size of the pinhole improves the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

Wandell, Foundations of Vision, Sinauer, 1995

The reason for lenses

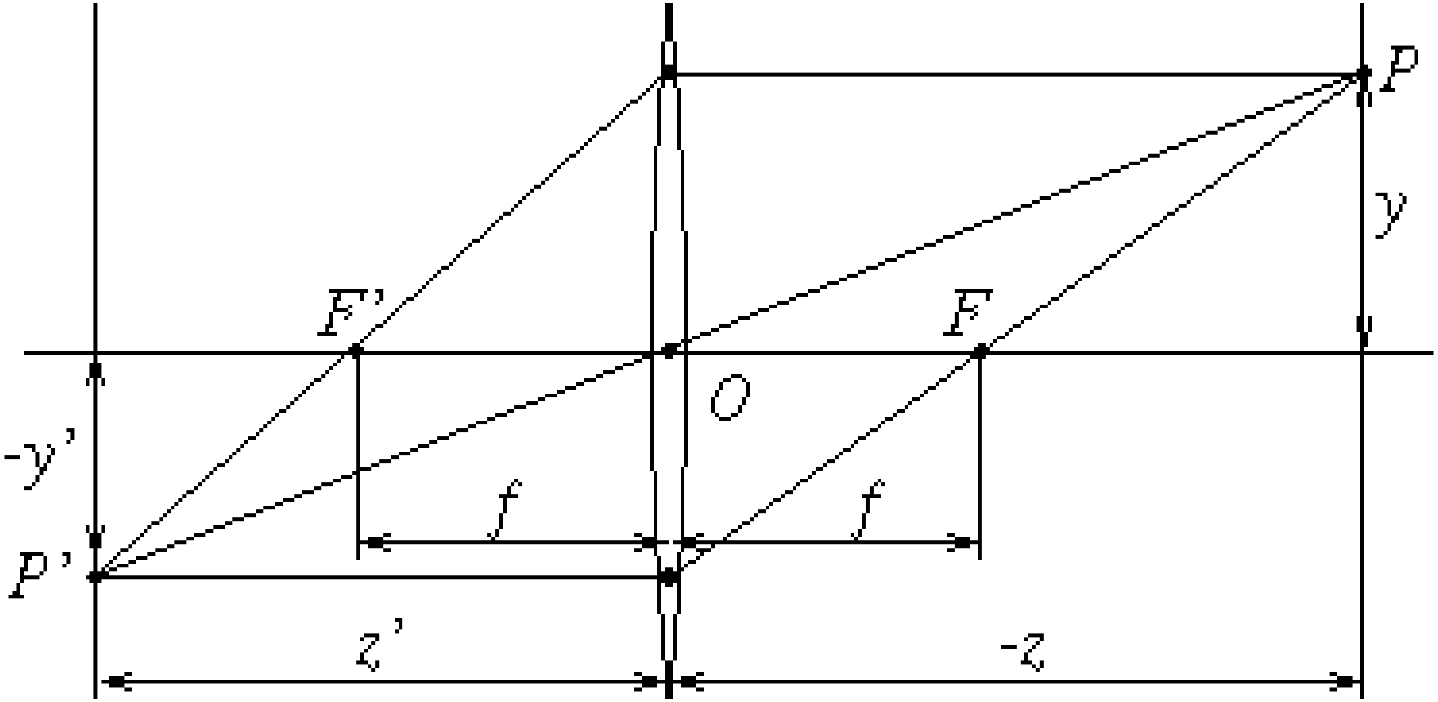


Water glass refraction



http://data.pg2k.hd.org/_exhibits/natural-science/cat-black-and-white-domestic-short-hair-DSH-with-nose-in-glass-of-water-on-bedside-table-tweaked-mono-1-AJHD.jpg

The thin lens, first order optics



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

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

All rays through P also pass through P', but only for points at -z: "*depth of field*".

Forsyth&Ponce

More accurate models of real lenses

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

Thick lens

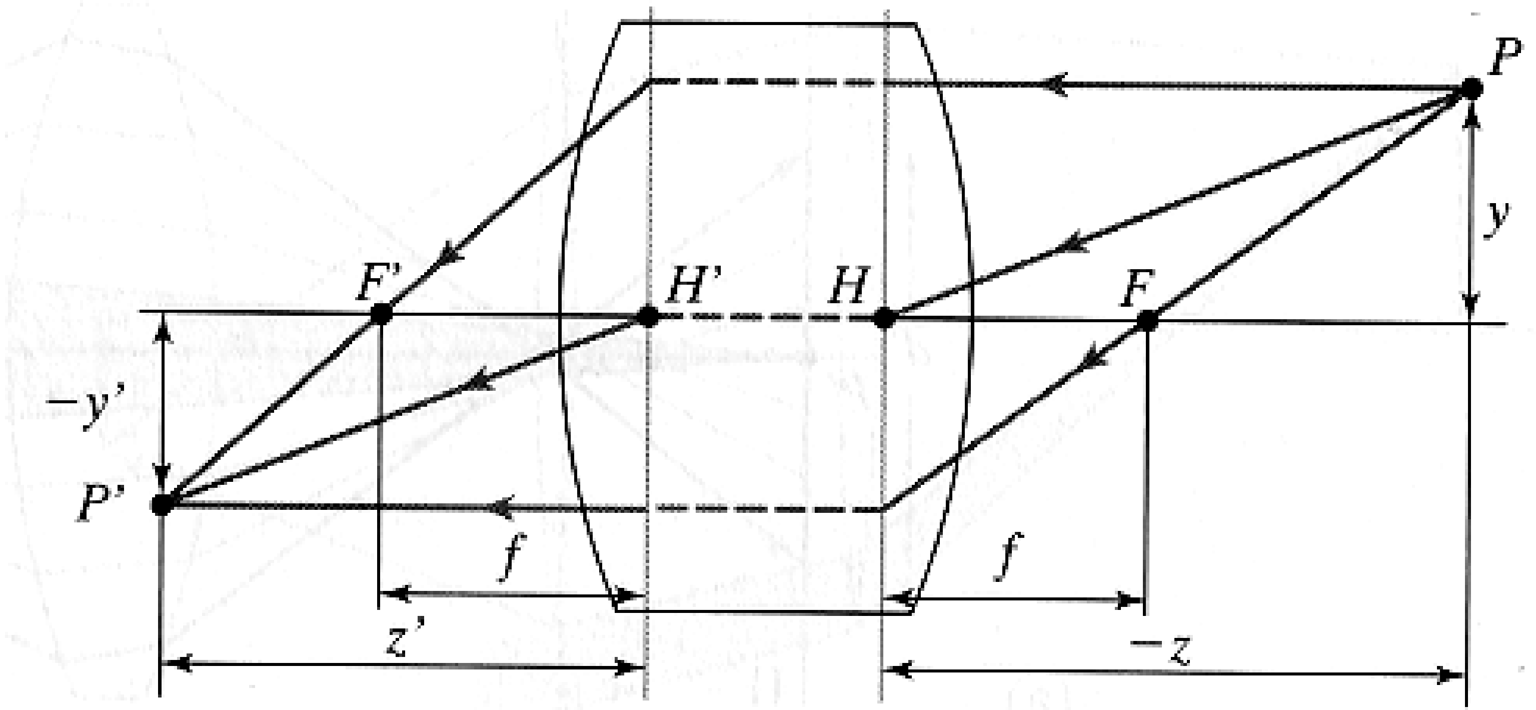
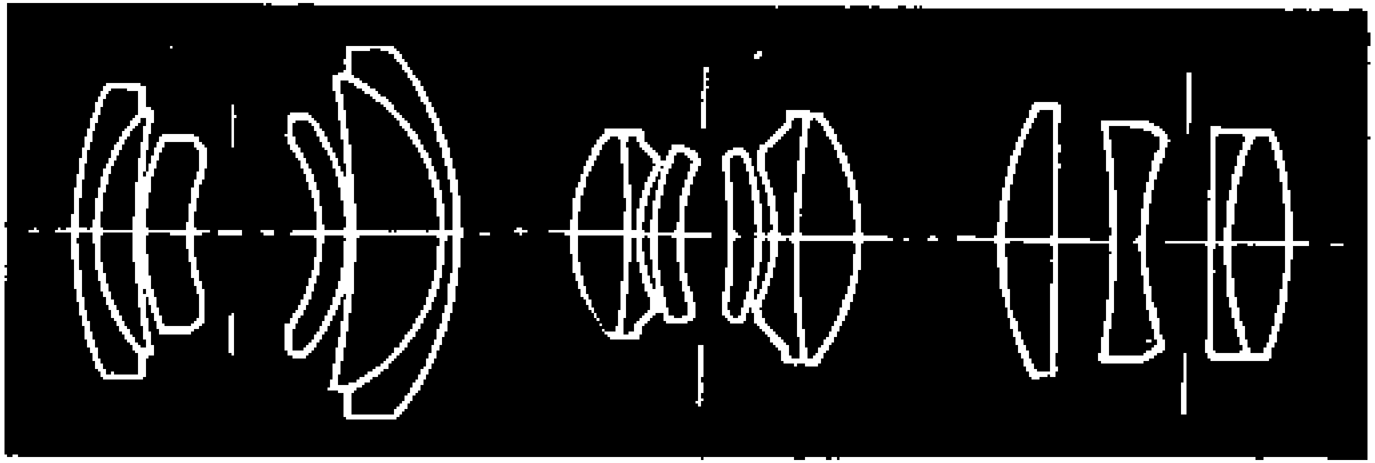


Figure 1.11 A simple thick lens with two spherical surfaces.

Forsyth&Ponce

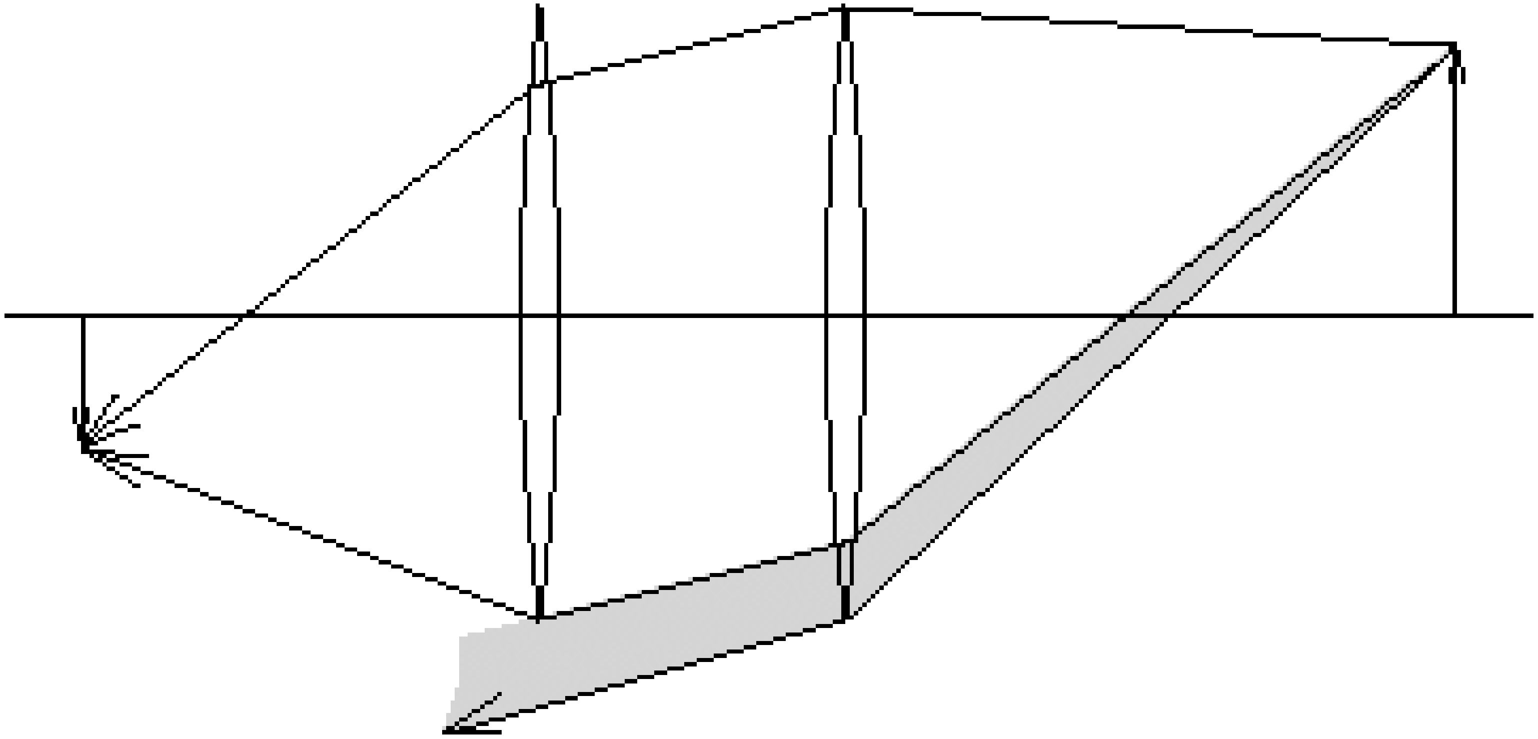
Lens systems



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

Forsyth&Ponce

Vignetting



Forsyth&Ponce

Chromatic aberration

(great for prisms, bad for lenses)

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 so far

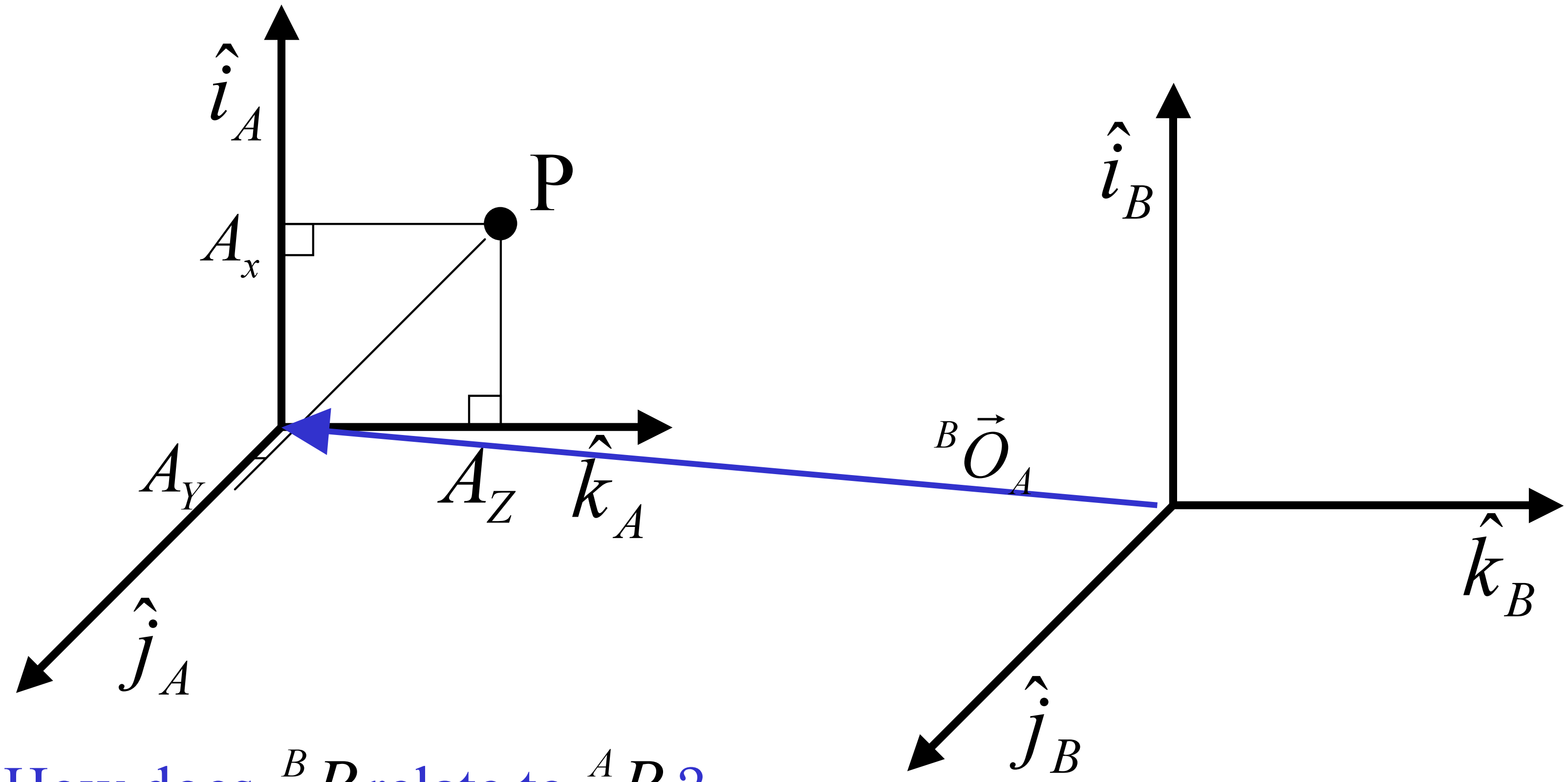
- Want to make images
- Pinhole camera models the geometry of perspective projection
- Lenses make it work in practice
- Models for lenses
  - Thin lens, spherical surfaces, first order optics
  - Thick lens, higher-order optics, vignetting.

Some background material…

- Rigid motion: translation and rotation
- Homogenous coordinates

Translation

$$ {}^{A}P = \begin{pmatrix} A_x \\ A_Y \\ A_Z \end{pmatrix} \qquad {}^{B}P = \begin{pmatrix} B_x \\ B_Y \\ B_Z \end{pmatrix} $$

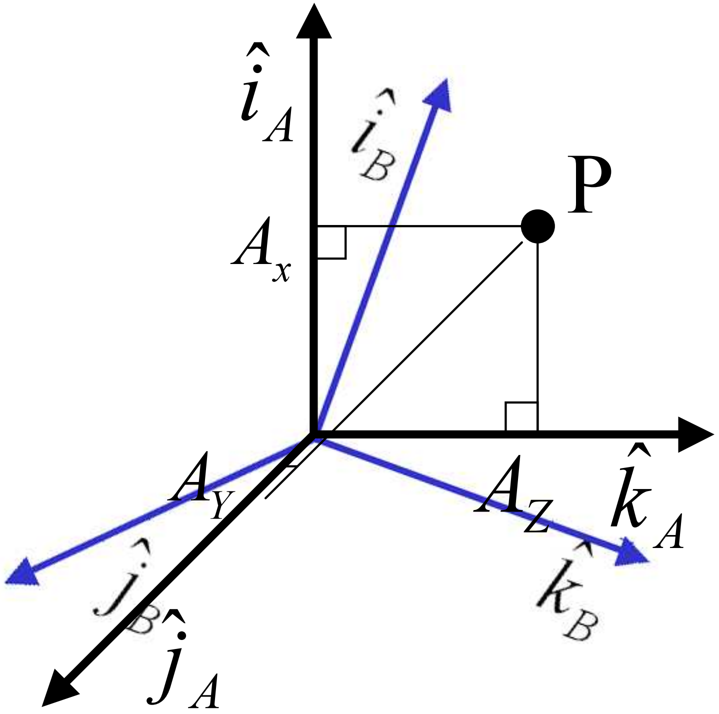


How does ${}^{B}P$ relate to ${}^{A}P$?

$$ {}^{B}P = {}^{A}P + {}^{B}O_A $$

Rotation

$$ {}^{A}P = \begin{pmatrix} A_x \\ A_Y \\ A_Z \end{pmatrix} \qquad {}^{B}P = \begin{pmatrix} B_x \\ B_Y \\ B_Z \end{pmatrix} $$

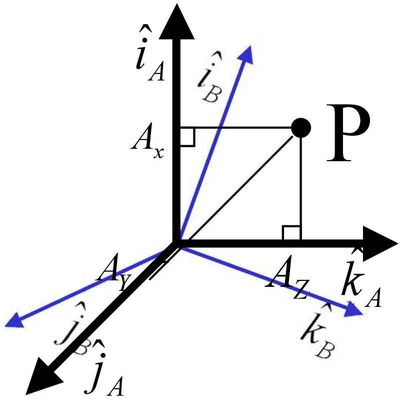


How does ${}^{B}P$ relate to ${}^{A}P$?

$$ {}^{B}P = {}^{B}_{A}R \; {}^{A}P $$

Find the rotation matrix

Project $\overrightarrow{OP} = \begin{pmatrix} \hat{i}_A & \hat{j}_A & \hat{k}_A \end{pmatrix} \begin{pmatrix} A_X \\ A_Y \\ A_Z \end{pmatrix}$



onto the B frame’s coordinate axes.

$$\begin{pmatrix} B_X \\ B_Y \\ B_Z \end{pmatrix} = \begin{pmatrix} \hat{i}_B \bullet \hat{i}_A A_X & \hat{i}_B \bullet \hat{j}_A A_Y & \hat{i}_B \bullet \hat{k}_A A_Z \\ \hat{j}_B \bullet \hat{i}_A A_X & \hat{j}_B \bullet \hat{j}_A A_Y & \hat{j}_B \bullet \hat{k}_A A_Z \\ \hat{k}_B \bullet \hat{i}_A A_X & \hat{k}_B \bullet \hat{j}_A A_Y & \hat{k}_B \bullet \hat{k}_A A_Z \end{pmatrix}$$

Rotation matrix

this

$$\begin{pmatrix} B_X \\ B_Y \\ B_Z \end{pmatrix} = \begin{pmatrix} \hat{i}_B \bullet \hat{i}_A A_X & \hat{i}_B \bullet \hat{j}_A A_Y & \hat{i}_B \bullet \hat{k}_A A_Z \\ \hat{j}_B \bullet \hat{i}_A A_X & \hat{j}_B \bullet \hat{j}_A A_Y & \hat{j}_B \bullet \hat{k}_A A_Z \\ \hat{k}_B \bullet \hat{i}_A A_X & \hat{k}_B \bullet \hat{j}_A A_Y & \hat{k}_B \bullet \hat{k}_A A_Z \end{pmatrix}$$

implies

$$^{B}P = {}^{B}_{A}R \; {}^{A}P$$

where

$$^{B}_{A}R = \begin{pmatrix} \hat{i}_B \bullet \hat{i}_A & \hat{i}_B \bullet \hat{j}_A & \hat{i}_B \bullet \hat{k}_A \\ \hat{j}_B \bullet \hat{i}_A & \hat{j}_B \bullet \hat{j}_A & \hat{j}_B \bullet \hat{k}_A \\ \hat{k}_B \bullet \hat{i}_A & \hat{k}_B \bullet \hat{j}_A & \hat{k}_B \bullet \hat{k}_A \end{pmatrix}$$

Translation and rotation

Let's write $${}^{B}P = {}^{B}_{A}R \; {}^{A}P + {}^{B}O_{A}$$

as a single matrix equation:

$$\begin{pmatrix} B_X \\ B_Y \\ B_Z \\ 1 \end{pmatrix} = \begin{pmatrix} - & - & - & | \\ - & {}^{B}_{A}R & - & {}^{B}O_{A} \\ - & - & - & | \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} A_X \\ A_Y \\ A_Z \\ 1 \end{pmatrix}$$

Homogenous coordinates

- Add an extra coordinate and use an equivalence relation
- for 3D
  - equivalence relation k*(X,Y,Z,T) is the same as (X,Y,Z,T)

- Motivation
  - Possible to write the action of a perspective camera as a matrix

Homogenous/non-homogenous transformations for a 3-d point

- From non-homogenous to homogenous coordinates: add 1 as the 4th coordinate, ie

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

- From homogenous to non-homogenous coordinates: divide 1st 3 coordinates by the 4th, ie

$$\begin{pmatrix} x \\ y \\ z \\ T \end{pmatrix} \rightarrow \frac{1}{T} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Homogenous/non-homogenous transformations for a 2-d point

- From non-homogenous to homogenous coordinates: add 1 as the 3rd coordinate, ie

$$\begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

- From homogenous to non-homogenous coordinates: divide 1st 2 coordinates by the 3rd, ie

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \rightarrow \frac{1}{z}\begin{pmatrix} x \\ y \end{pmatrix}$$

The camera matrix, in homogenous coordinates

- Turn previous expression into HC's
  - HC's for point in 3D point are (X,Y,Z,T)
  - HC's for point in image are (U,V,W)

$$\begin{pmatrix} X \\ Y \\ \frac{Z}{f} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

$$\begin{pmatrix} X \\ Y \\ \frac{Z}{f} \end{pmatrix} \rightarrow \frac{f}{Z} \begin{pmatrix} X \\ Y \end{pmatrix}$$

HC Non-HC

What about an orthographic camera?

The projection matrix for orthographic projection, homogenous coordinates

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

$$= \begin{pmatrix} X \\ Y \\ T \end{pmatrix} \rightarrow \frac{1}{T} \begin{pmatrix} X \\ Y \end{pmatrix}$$

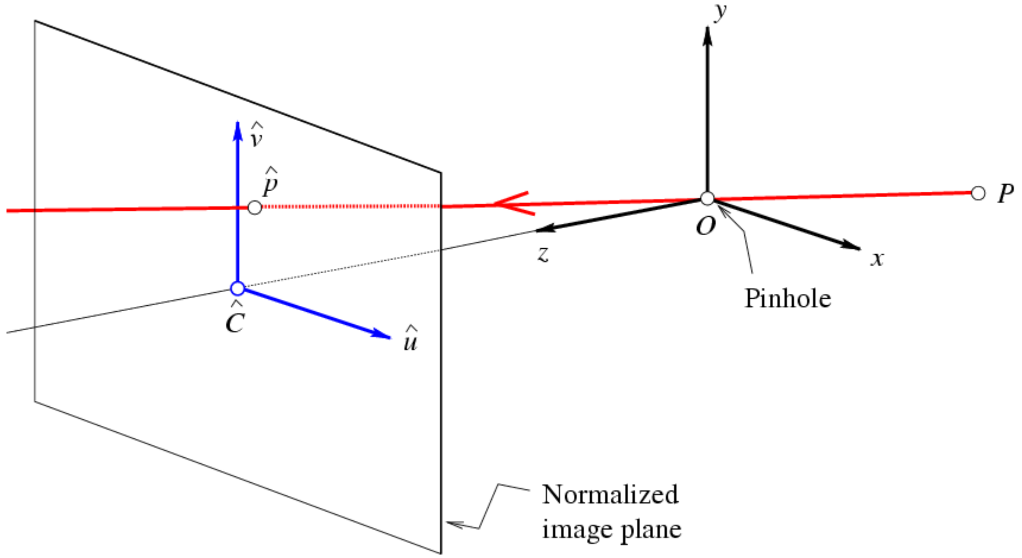
HC Non-HC

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.*
- (Relationship between intensities in the world and intensities in the image: *photometric camera calibration*, not covered in this course, see 6.801 or text)

Intrinsic parameters



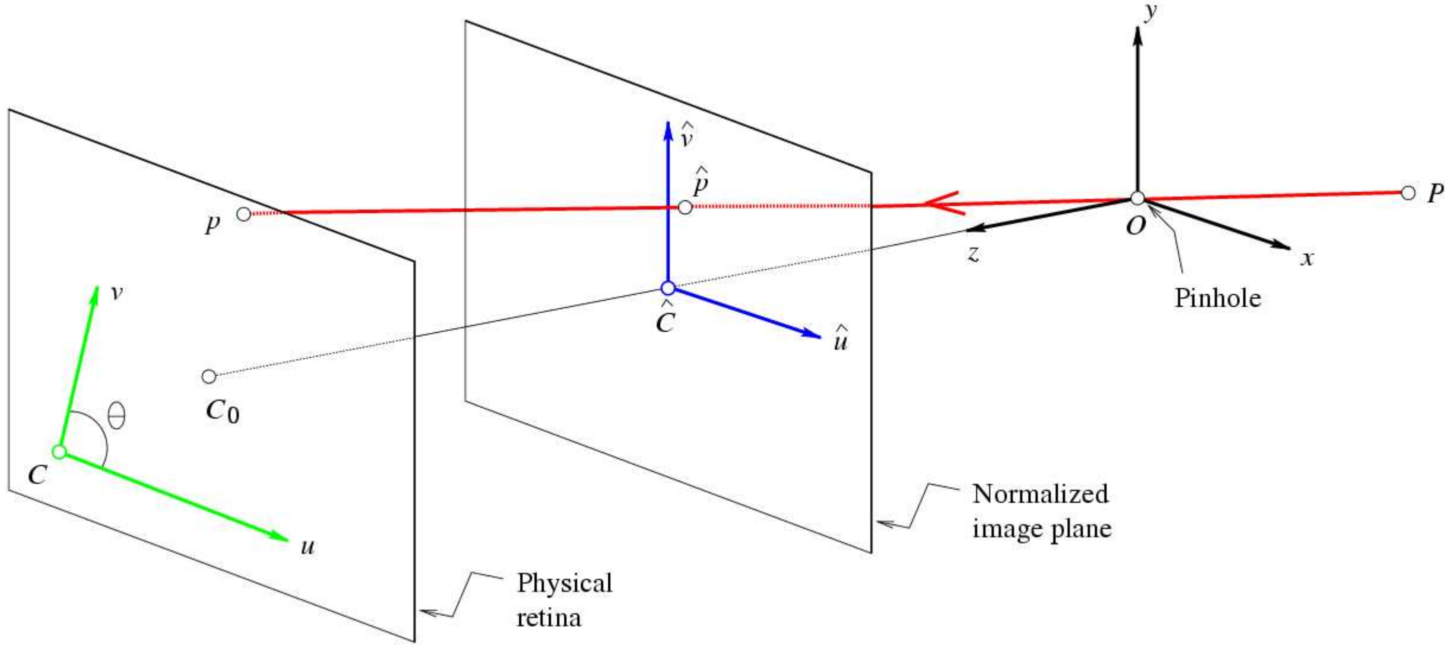
Forsyth&Ponce

Perspective projection

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

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

Intrinsic parameters

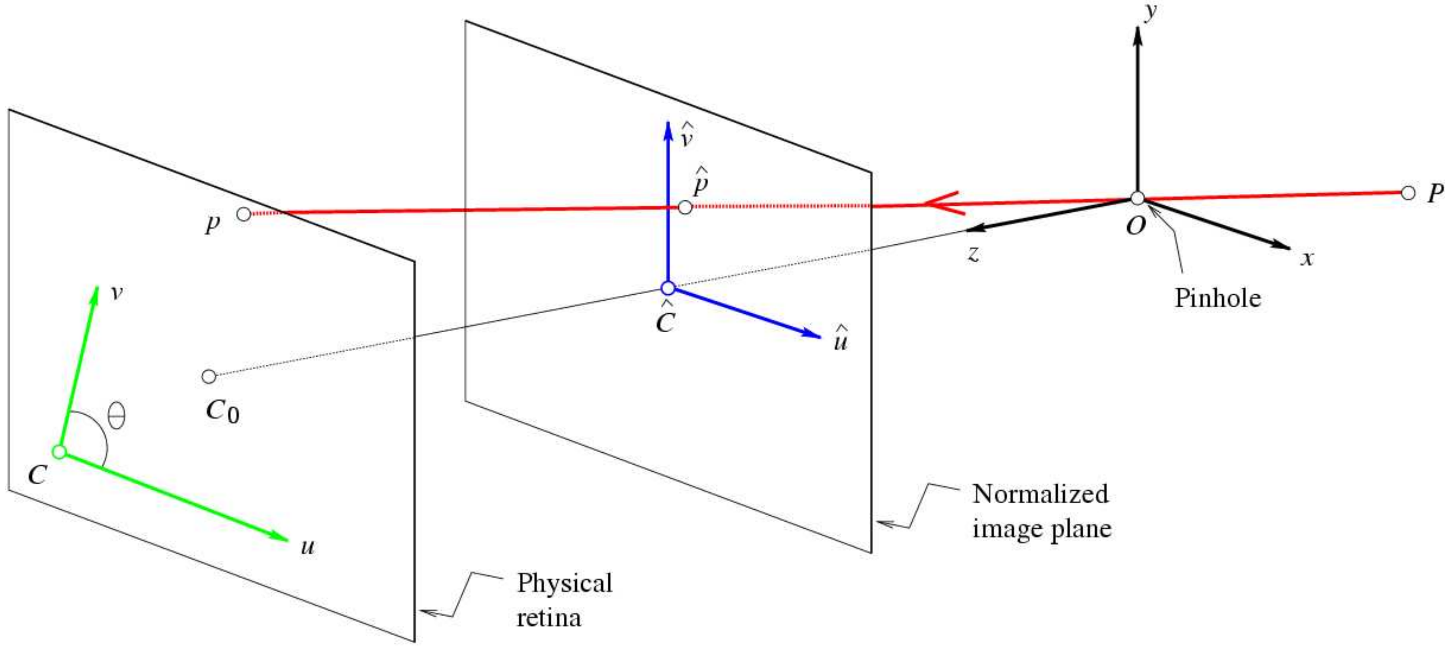


But “pixels” are in some arbitrary spatial units…

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

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

Intrinsic parameters

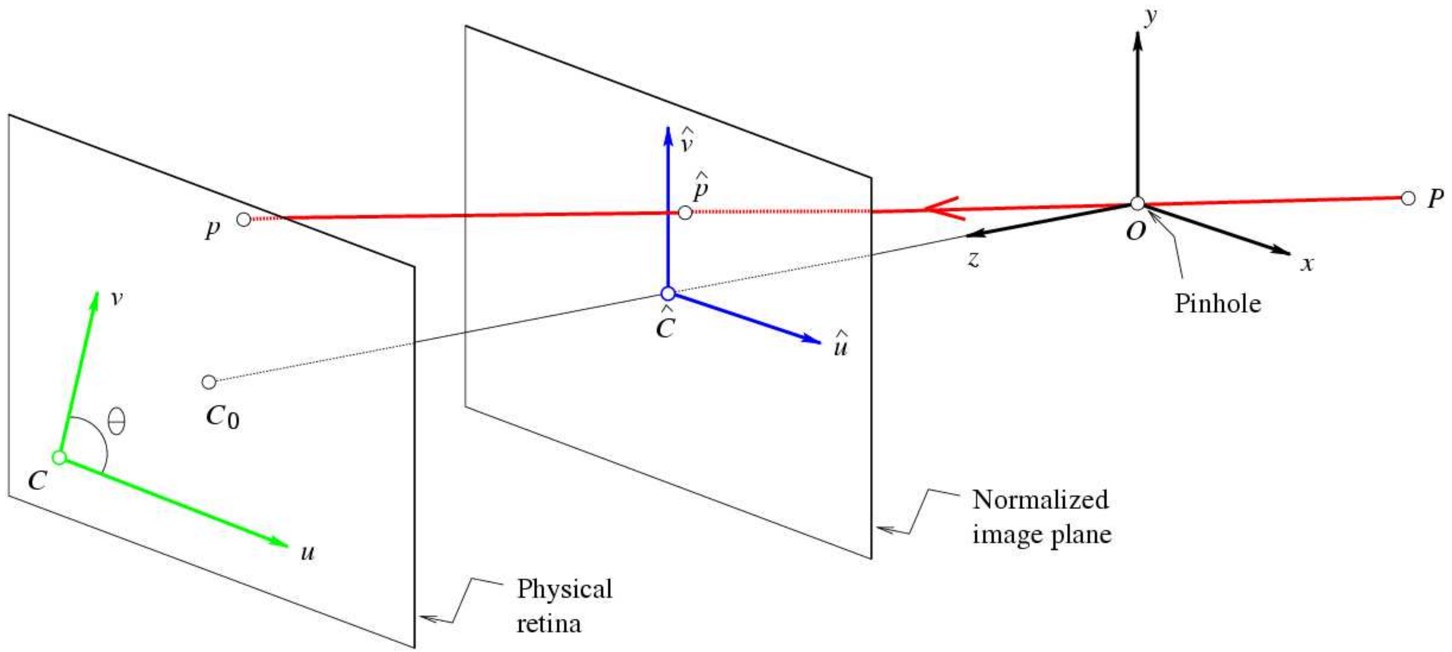


But "pixels" are in some arbitrary spatial units

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

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

Intrinsic parameters

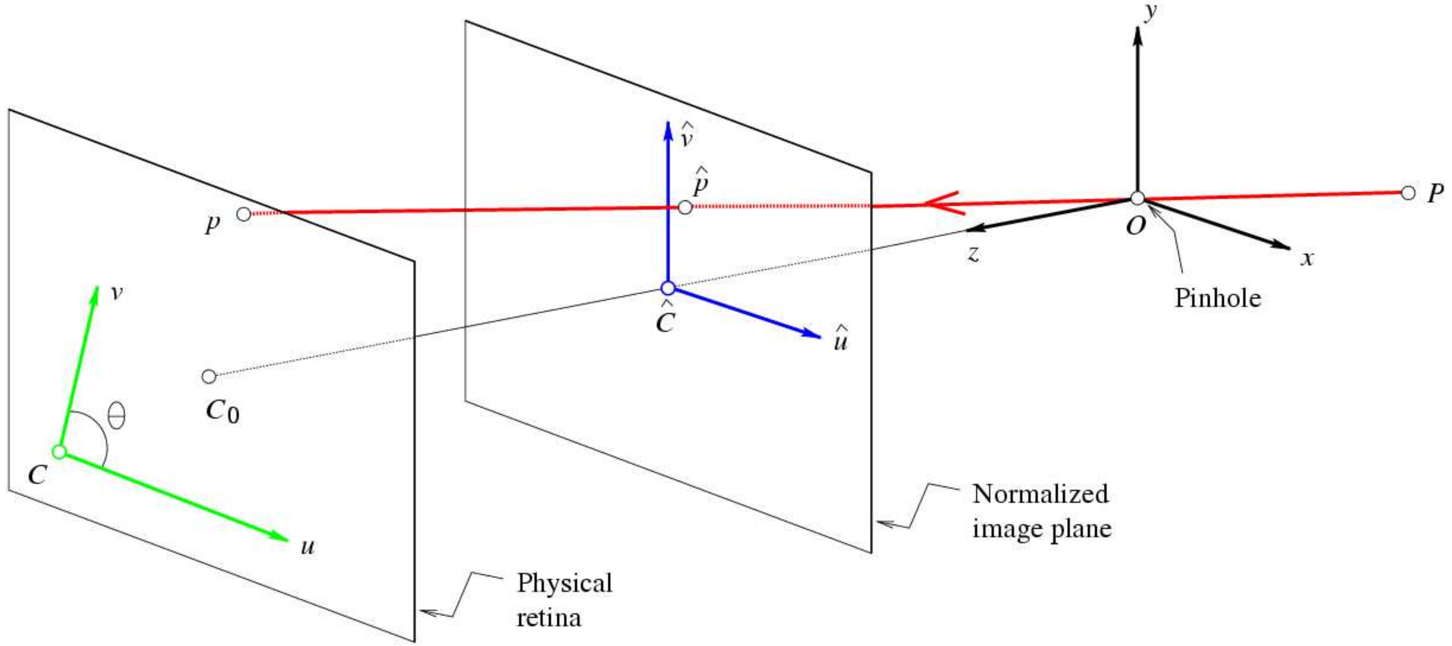


Maybe pixels are not square…

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

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

Intrinsic parameters

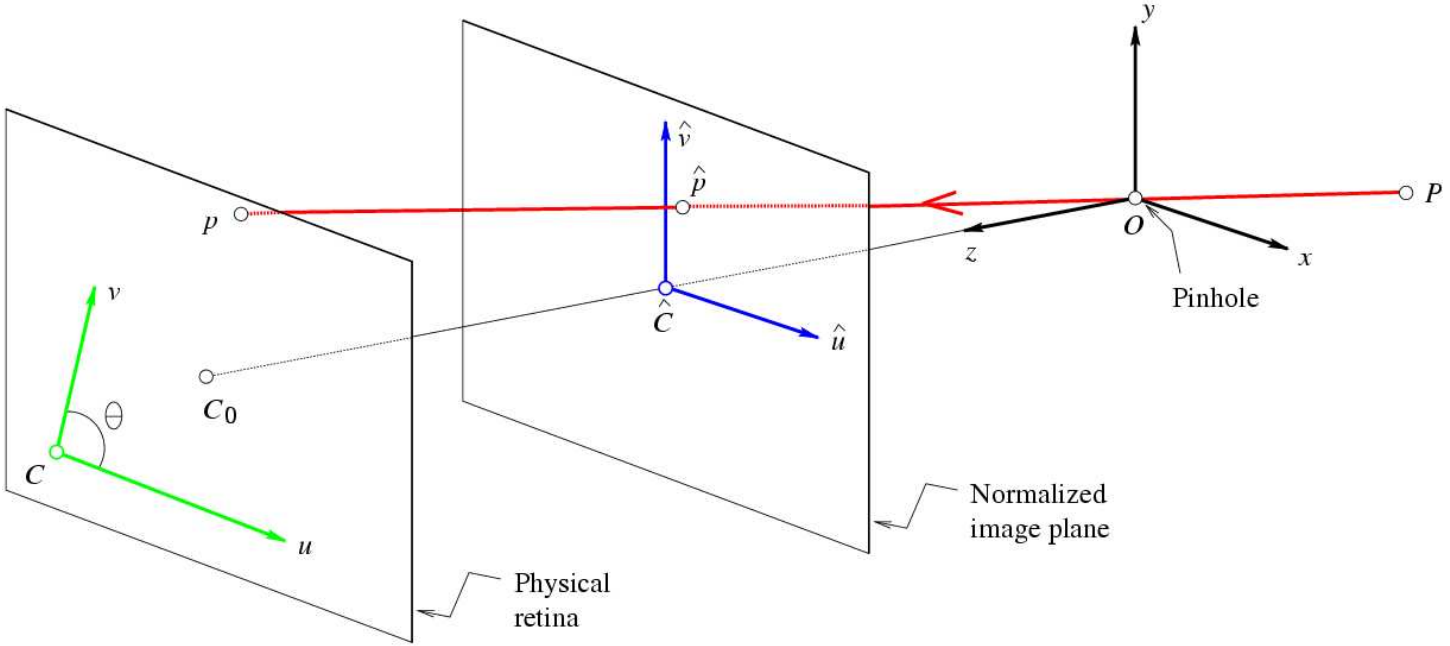


Maybe pixels are not square

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

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

Intrinsic parameters

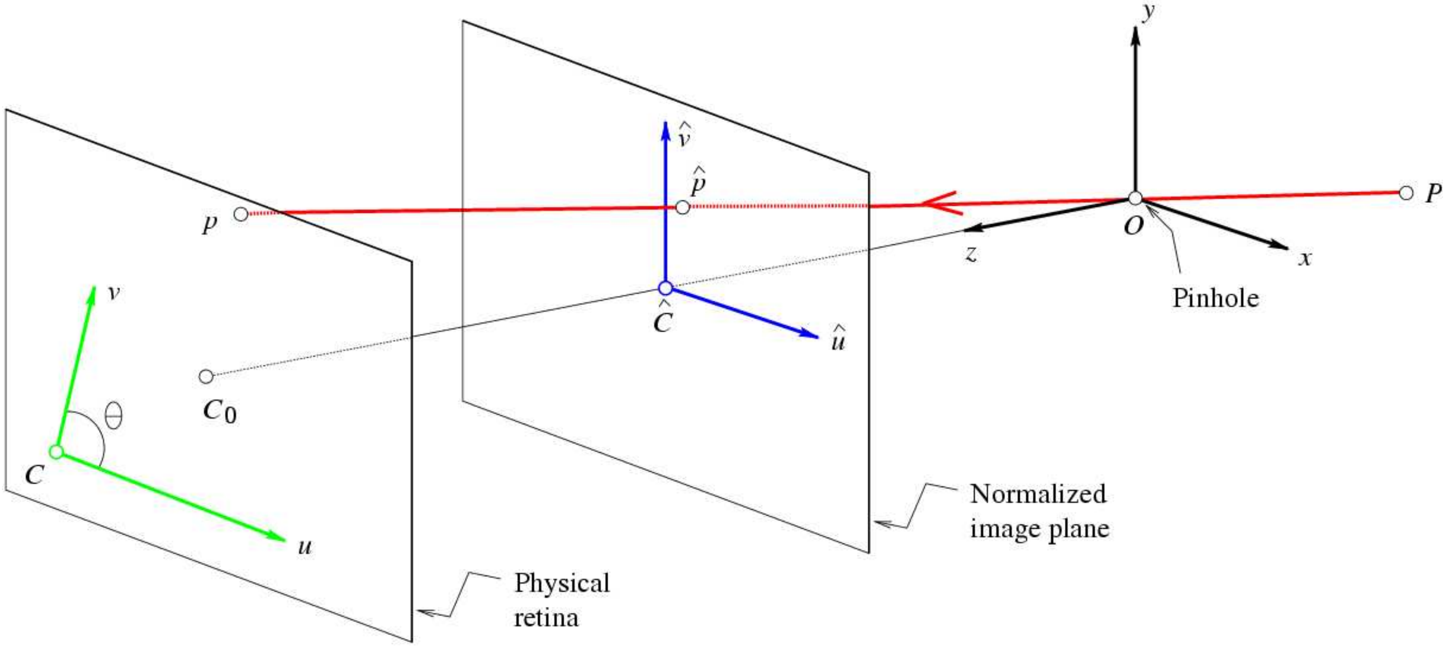


We don’t know the origin of our camera pixel coordinates…

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

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

Intrinsic parameters

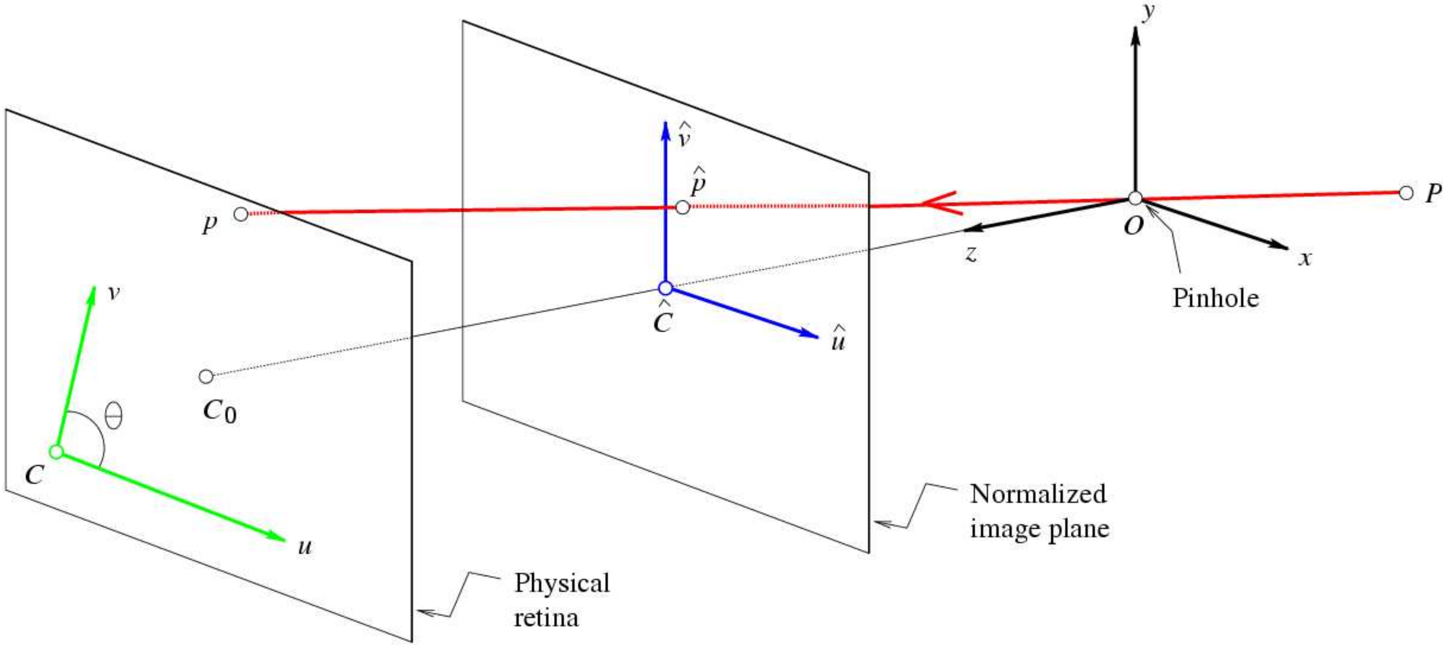


We don’t know the origin of our camera pixel coordinates

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

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

Intrinsic parameters

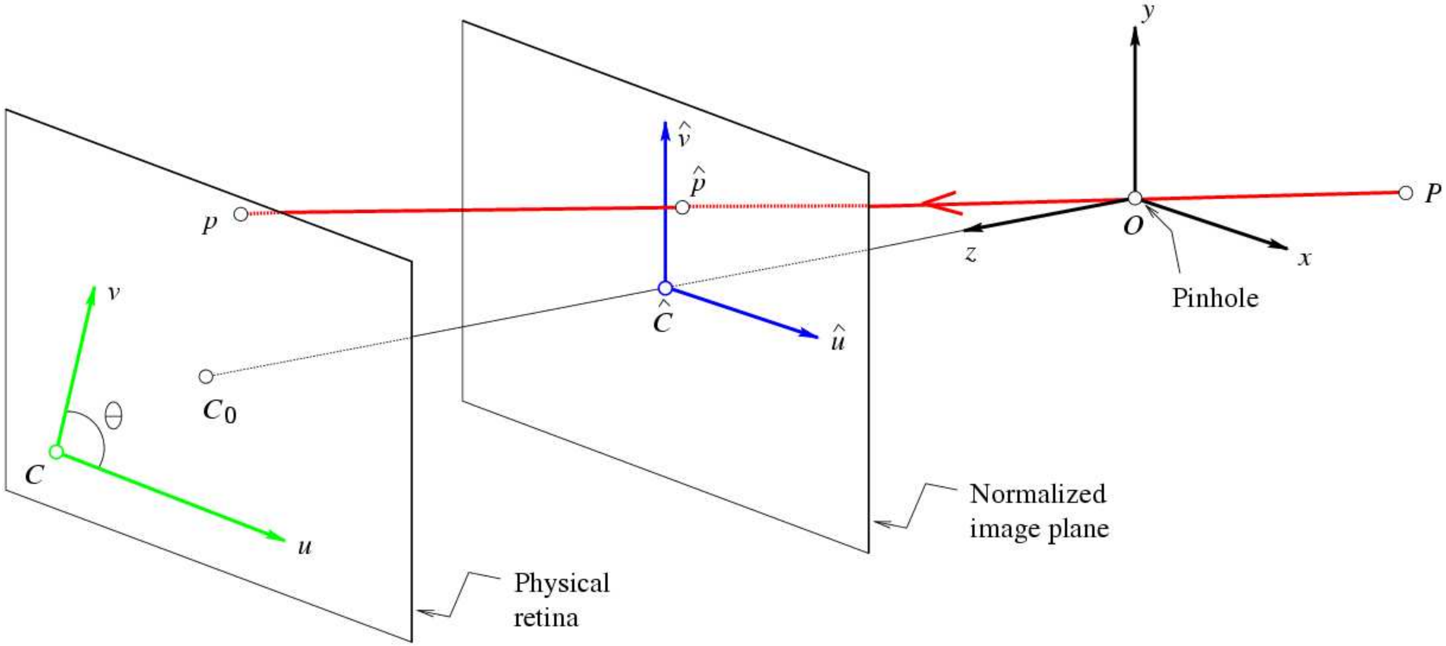


May be skew between camera pixel axes…

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

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

Intrinsic parameters

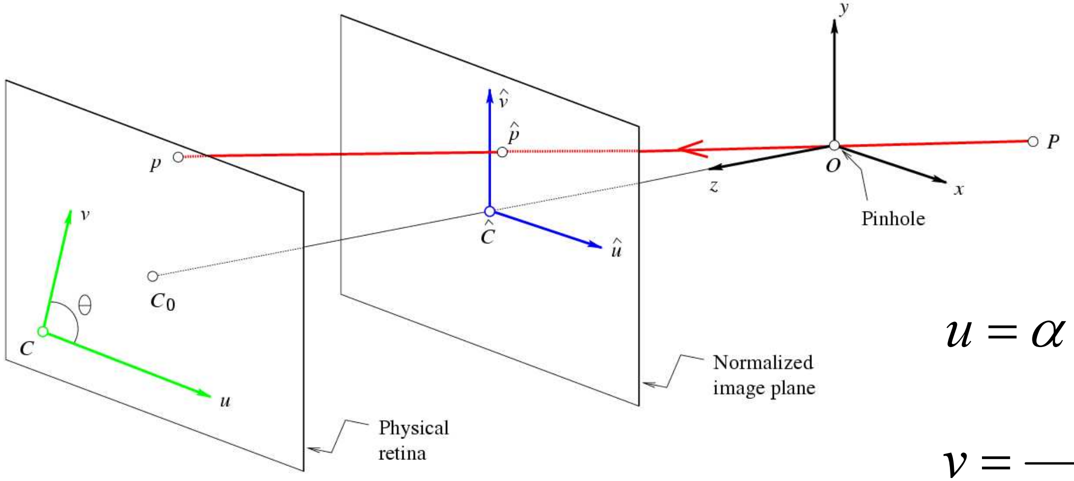


May be skew between camera pixel axes

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

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

Intrinsic parameters



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

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

Using homogenous coordinates, we can write this as:

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

or:

$$\boxed{\vec{p} = \frac{1}{z} \begin{pmatrix} K & \vec{0} \end{pmatrix} \vec{P}}$$

Extrinsic parameters: translation and rotation of camera frame

$$ {}^{C}P = {}^{C}_{W}R \; {}^{W}P + {}^{C}O_{W} $$

Non-homogeneous coordinates

$$ \begin{pmatrix} C_X \\ C_Y \\ C_Z \\ 1 \end{pmatrix} = \begin{pmatrix} - & - & - & | \\ - & {}^{C}_{W}R & - & {}^{C}O_{W} \\ - & - & - & | \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} W_X \\ W_Y \\ W_Z \\ 1 \end{pmatrix} $$

Homogeneous coordinates

$$ \begin{pmatrix} {}^{C}P \\ 1 \end{pmatrix} = \begin{pmatrix} {}^{C}_{W}\mathcal{R} & {}^{C}O_{W} \\ \mathbf{0}^{T} & 1 \end{pmatrix} \begin{pmatrix} {}^{W}P \\ 1 \end{pmatrix} $$

Block matrix form

Combining extrinsic and intrinsic calibration parameters

$$\vec{p} = \frac{1}{z}\begin{pmatrix} K & \vec{0} \end{pmatrix}\vec{P}$$

Intrinsic

$${}^{C}P = {}^{C}_{W}R\;{}^{W}P + {}^{C}O_{W}$$

Extrinsic

$$\vec{p} = \frac{1}{z}\,K\begin{pmatrix} {}^{C}_{W}R & {}^{C}O_{W} \end{pmatrix}\vec{P}$$

$$\vec{p} = \frac{1}{z}\,M\;\vec{P}$$

Forsyth&Ponce

Other ways to write the same equation

pixel coordinates

world coordinates

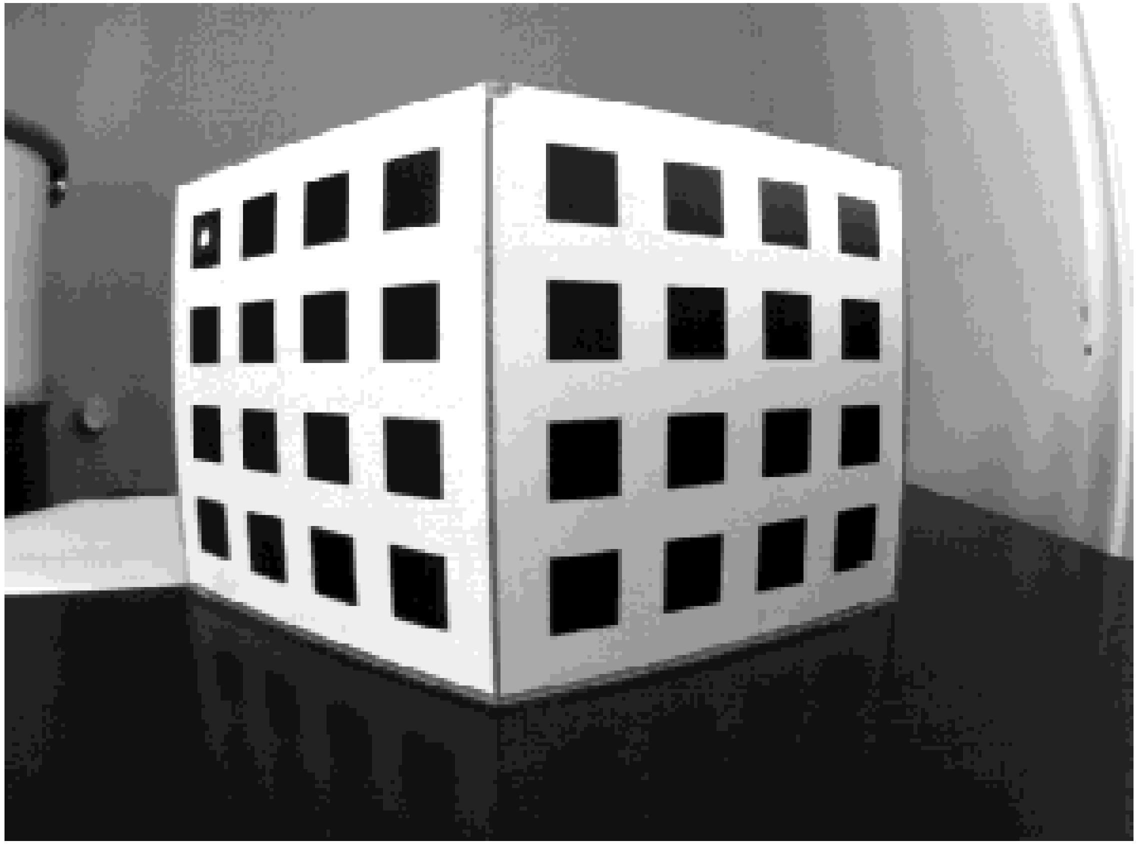
$$\vec{p} = \frac{1}{z} M\vec{P}$$

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \frac{1}{z}\begin{pmatrix} . & m_1^T & . & . \\ . & m_2^T & . & . \\ . & m_3^T & . & . \end{pmatrix}\begin{pmatrix} W_x \\ W_y \\ W_z \\ 1 \end{pmatrix}$$

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

z is in the *camera* coordinate system, but we can solve for that, since $1 = \frac{m_3 \cdot \vec{P}}{z}$, leading to:

Calibration target



The Opti-CAL Calibration Target Image

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

Camera calibration

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

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

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

So for each feature point, i, we have:

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

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

Camera calibration

Stack all these measurements of i=1…n points

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

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

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:

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

Showing all the elements:

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

Camera calibration

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

P m = 0

We want to solve for the unit vector m (the stacked one) that minimizes $\left|Pm\right|^2$

The minimum eigenvector of the matrix $P^T P$ gives us that (see Forsyth&Ponce, 3.1)

Camera calibration

Once you have the M matrix, can recover the intrinsic and extrinsic parameters as in Forsyth&Ponce, sect. 3.2.2.