Computer Vision

CS-E4850, 5 study credits

Lecturer: Juho Kannala

Lecture 1a: Introduction

- Motivation:
  - What is computer vision?
  - Why to study computer vision?
- Specifics of this course
- Questions

- Today also: Lecture 1b - Image formation

A bit about me

Juho Kannala
Assistant Professor of Computer Vision

- MSc, TKK 2004
- PhD, University of Oulu 2010
- Professor at Aalto since 2016

- I got introduced to computer vision as a research assistant in 2000
- Working in the field since then, 11 years in Oulu (2004-2015)
- Examples of recent projects at https://users.aalto.fi/~kannalj1/

What is Computer Vision?

- What are examples of computer vision being used in the world?
- Why computer vision matters?

Acknowledgement: many slides from James Hays, Derek Hoiem, Svetlana Lazebnik, Steve Seitz, Alexei Efros, David Forsyth, Silvio Savarese and others (detailed credits on individual slides)

Computer Vision

Make computers understand images and video.

What kind of scene?

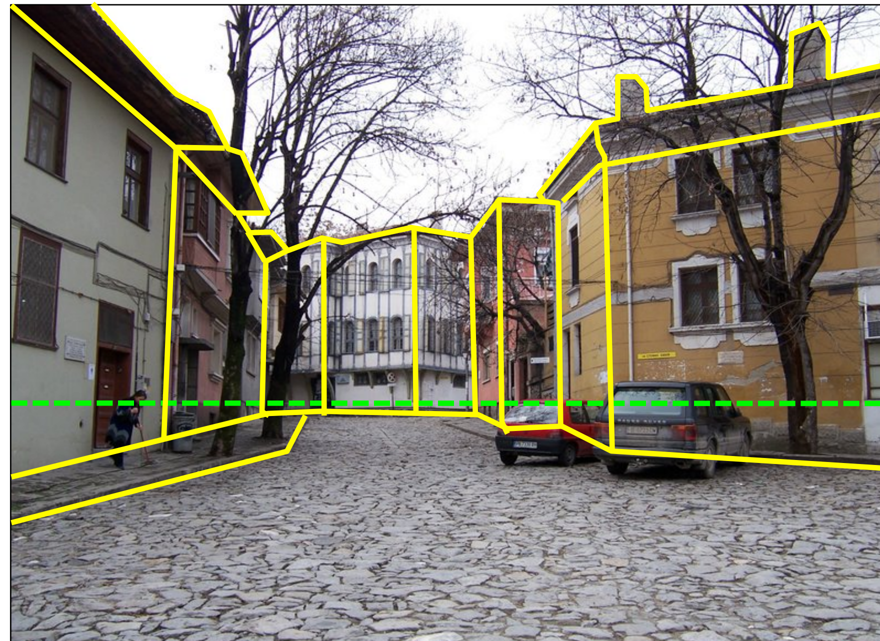
Where are the cars?

How far are the buildings?

…

Slide adapted from James Hays

What kind of information can be extracted from an image?



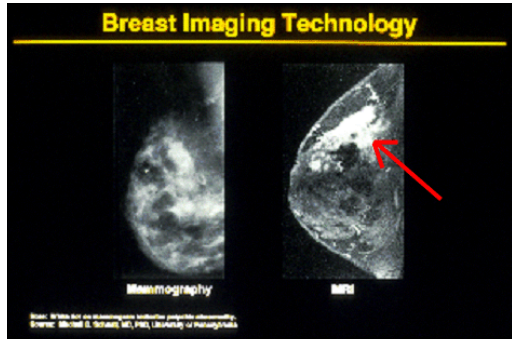
Semantic information

Geometric information

Slide by Svetlana Lazebnik

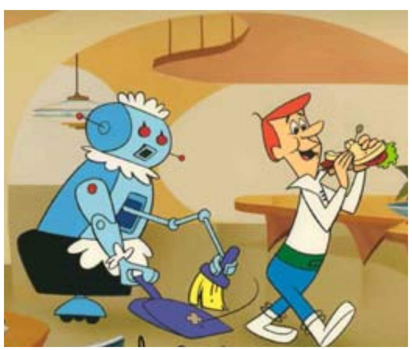
Computer vision matters

Safety



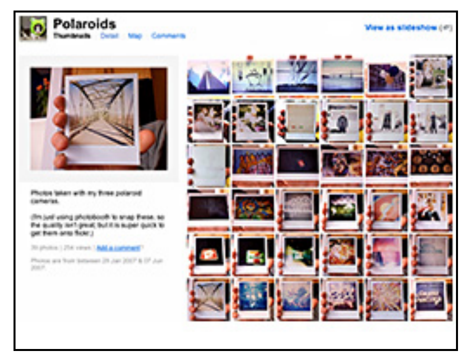
Health

Security



Comfort

Fun



Access

Slide by Derek Hoiem, James Hays

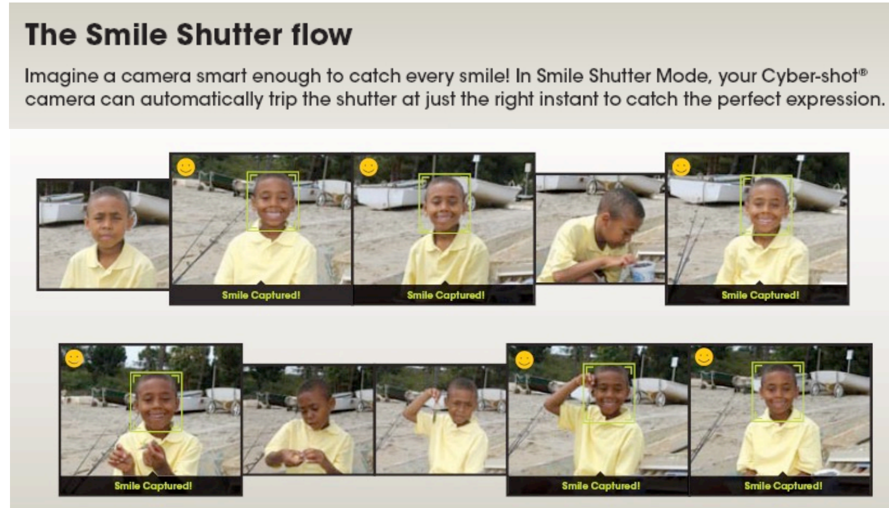
Successes of computer vision to date…

Several of the following slides by Svetlana Lazebnik and Derek Hoiem

Recognizing “simple” patterns

Faces

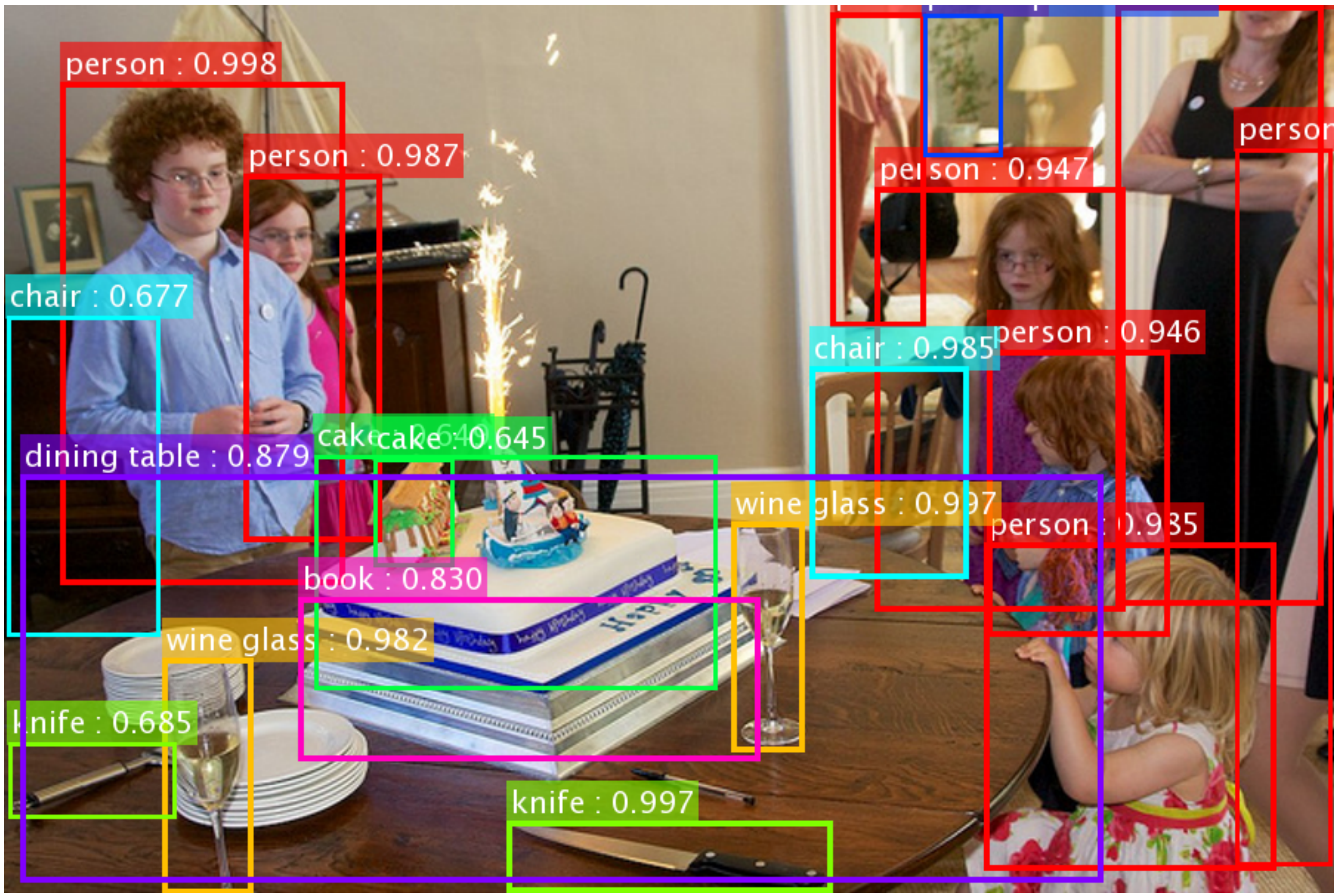

MPEGMOVIE VX
MENU
DISP
HOME
SONY


The Smile Shutter flow
Imagine a camera smart enough to catch every smile! In Smile Shutter Mode, your Cyber-shot® camera can automatically trip the shutter at just the right instant to catch the perfect expression.
Smile Captured!
Smile Captured!
Smile Captured!
Smile Captured!
Smile Captured!
Smile Captured!


iPhoto
LIBRARY
Events
Photos
Faces
Places
RECENT
Road Trip
Last 12 Months
Last Import
Flagged
Trash
ALBUMS
5-Star Photos
Our Family
MOBILEME GALLERY
Graduation
Italy
Surfing
FACEBOOK
Road Trip
All
Add Missing Face
Done
14 Faces in 1,023 photos


KFC

Object detection and recognition



Code available!
http://kaiminghe.com

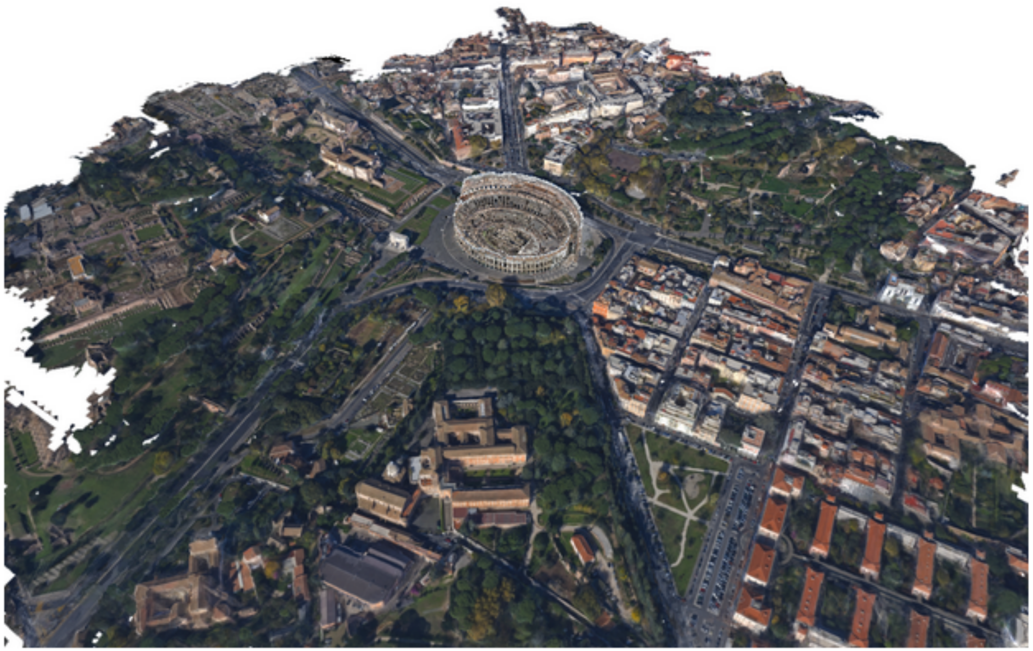
ResNet's object detection result on COCO

*the original image is from the COCO dataset

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Reconstruction: 3D from photo collections

Colosseum, Rome, Italy

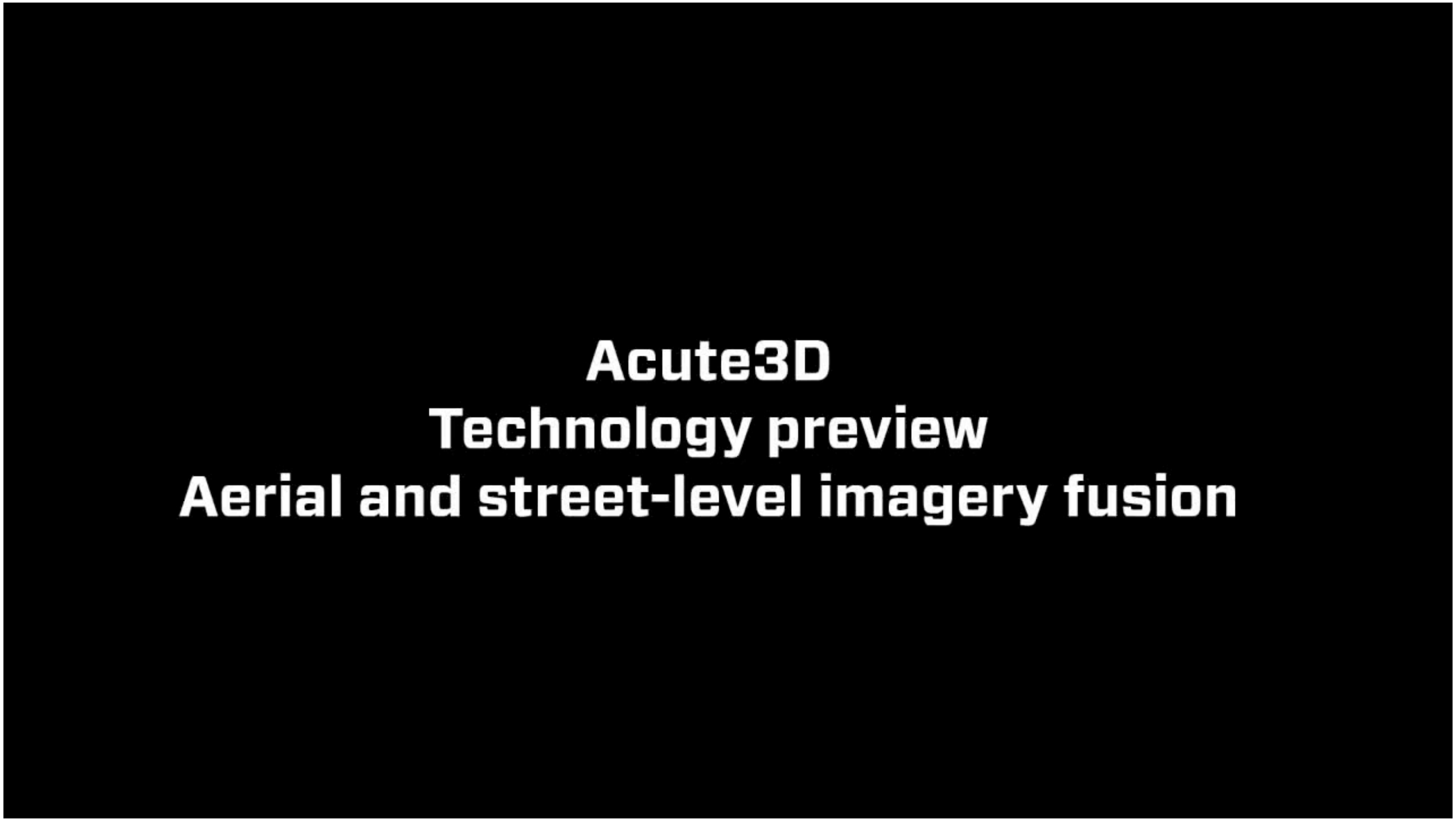


San Marco Square, Venice, Italy

Q. Shan, R. Adams, B. Curless, Y. Furukawa, and S. Seitz,
The Visual Turing Test for Scene Reconstruction, 3DV 2013

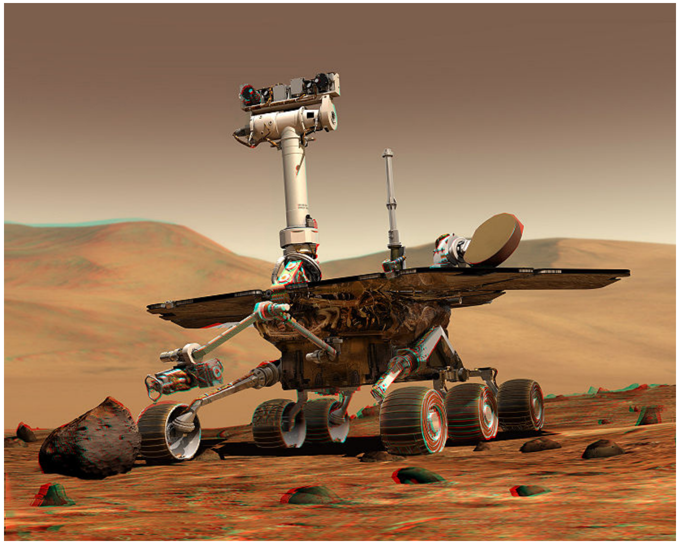
YouTube Video

A recent commercial 3D reconstruction system

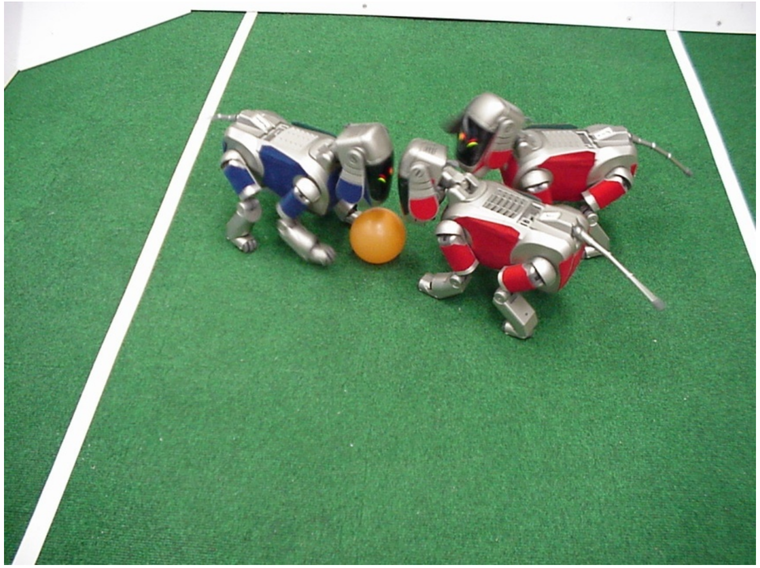


https://www.youtube.com/user/Acute3D

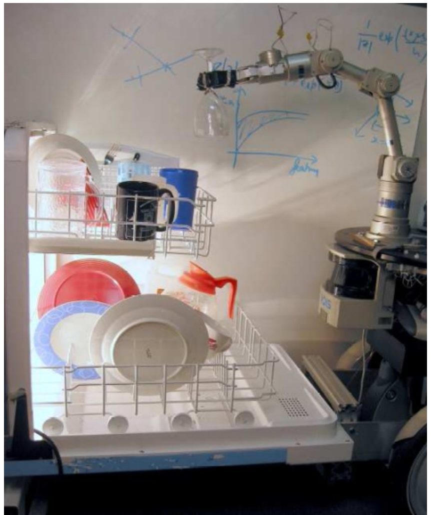
Mobile robots



NASA's Mars Spirit Rover
http://en.wikipedia.org/wiki/Spirit_rover
(For more, read "Computer Vision on Mars" by Matthies et al.)



http://www.robocup.org/

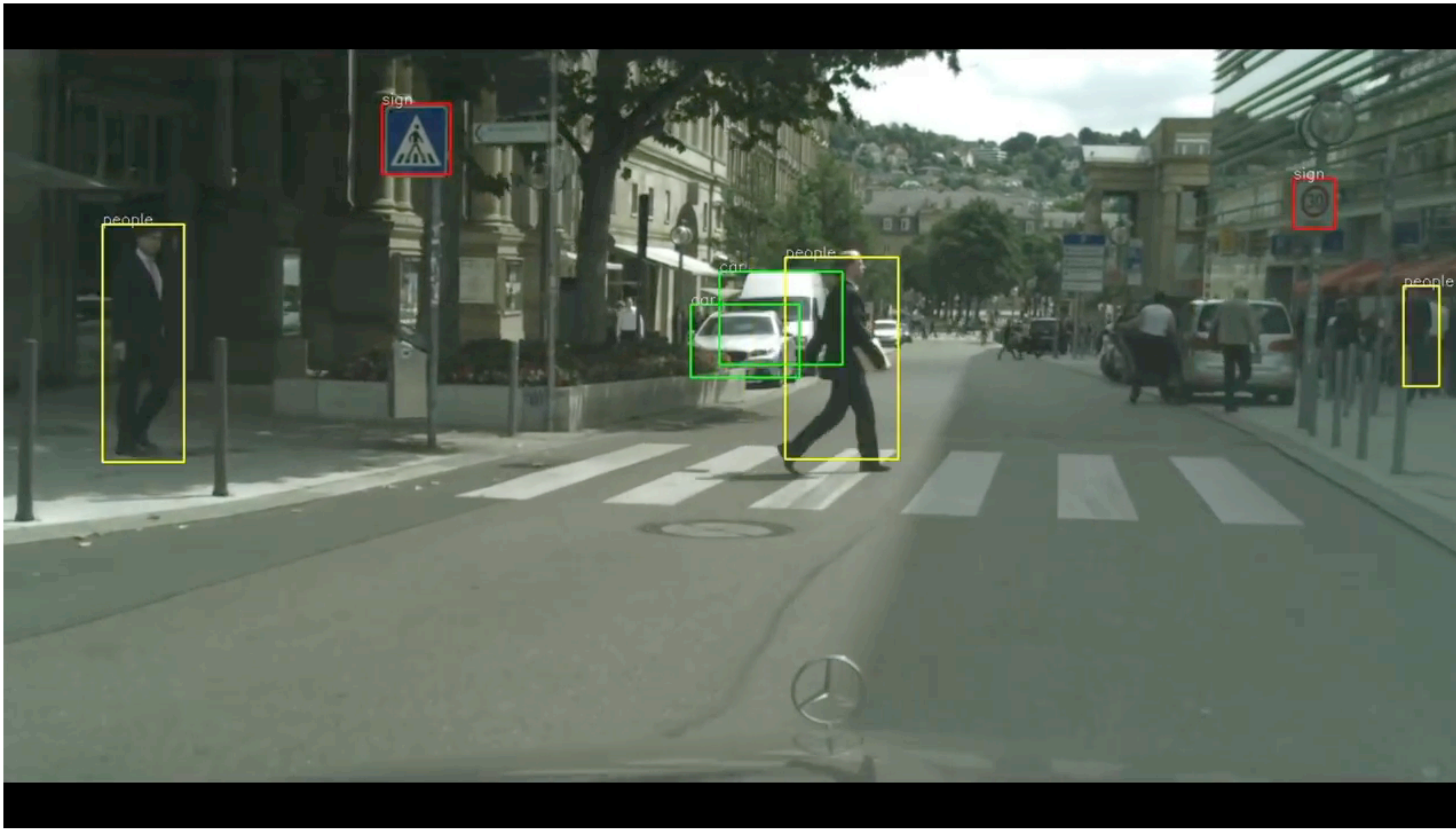


Saxena et al. 2008
STAIR at Stanford

Slide by James Hays

A recent legged robot

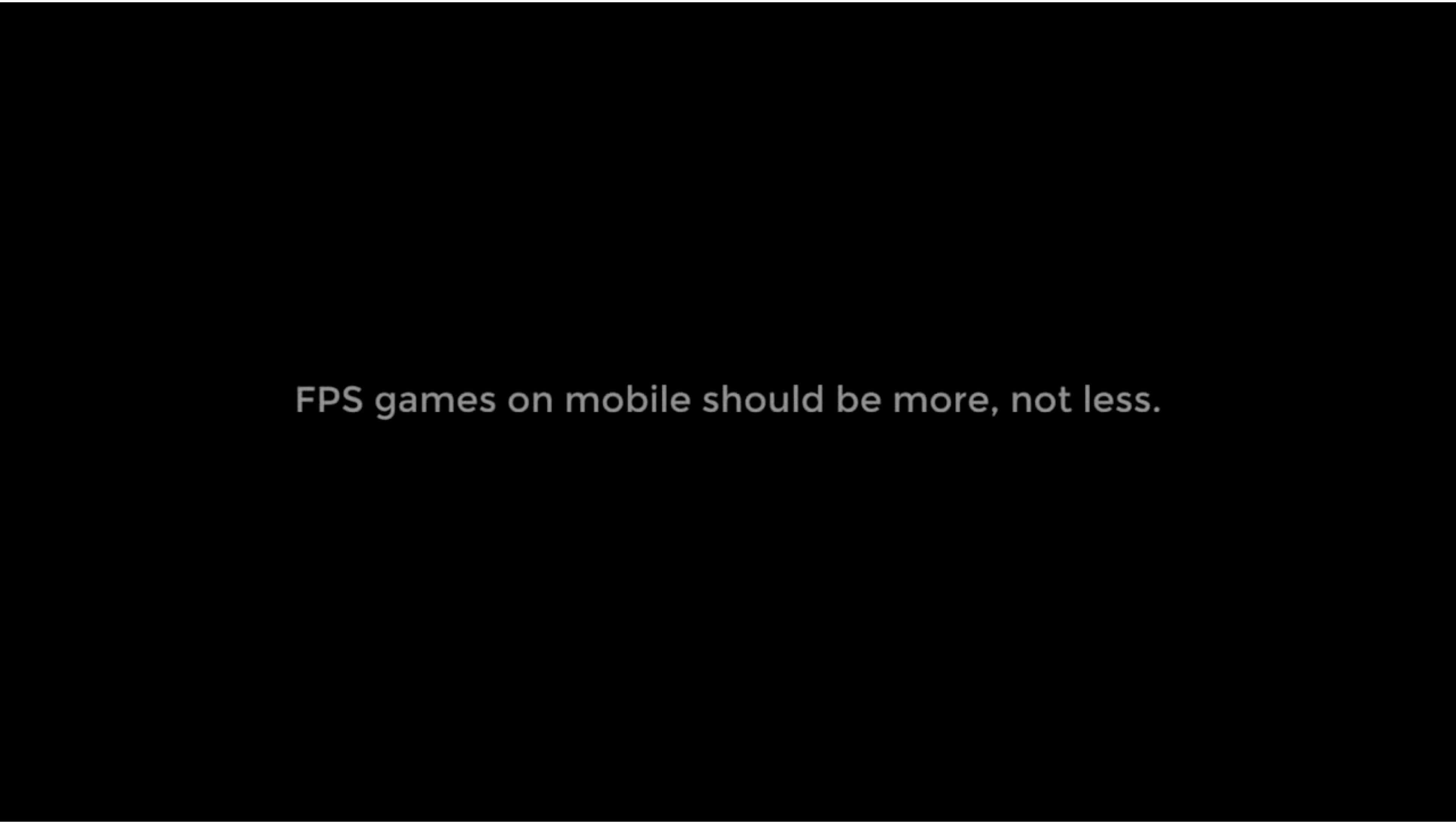
Self-driving cars (Nvidia @ CES 2016)



Visual odometry & SLAM

Youtube video link by 13thlab

Augmented Reality and Virtual Reality



Youtube video by 13thlab; see also Magic Leap, Oculus, Hololens, etc.

Current state of the art

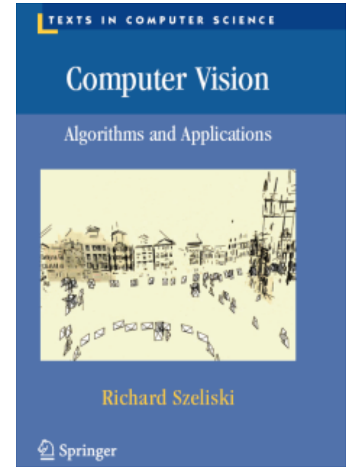
- You just saw examples of current systems
  - Many of these are less than 5 years old
- Active and rapidly changing research area
  - Many new applications in the next 5 years
- Many recent state-of-the-art methods are available as open-source software!
  - See papers from top conferences (e.g. CVPR, ECCV, ICCV, ICML, NIPS)

Also plenty of job opportunities!

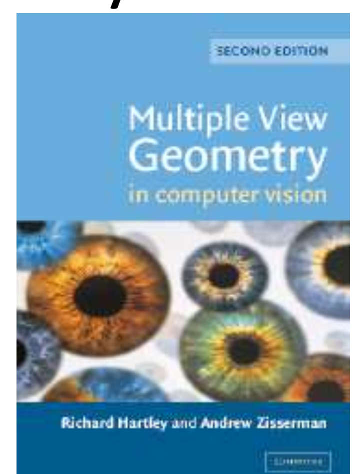
- Experts in computer vision and deep learning are needed in companies too
- Big Internet companies are investing heavily (Apple, Baidu, Google, Facebook, Microsoft,…) and car industry too (Tesla, Uber, Volvo, …)
- Recent activity also in Finland and Nordics, e.g.
  - http://m.kauppalehti.fi/uutiset/volvo-palkkaa-sadoittain-softainsinooreja/BVE7npxH
  - https://techcrunch.com/2016/10/28/magic-leap-goes-to-finland-in-pursuit-of-nordic-vr-and-ar-talent/
  - https://www.indeed.fi/Computer-Vision-jobs

Course textbooks

- Szeliski: Computer Vision
  - http://szeliski.org/Book/
  - Full-copy freely available



- Hartley & Zisserman: Multiple View Geometry in Computer Vision
  - http://www.robots.ox.ac.uk/~vgg/hzbook/
  - Available as e-book via Aalto Library



Other useful books

- Forsyth & Ponce: Computer Vision
  - http://courses.cs.washington.edu/courses/cse455/02wi/readings/book-7-revised-a-indx.pdf
- Gonzalez & Woods: Digital Image Processing
  - http://www.imageprocessingplace.com/

Course overview

Tentative course content (numbers refer to chapters in [Szeliski]):

- Image formation (2)
- Image processing (3)
- Feature detection & matching (4)
- Feature based alignment & image stitching (6,9)
- Dense motion estimation (8)
- Structure from motion (7)
- Stereo and 3D reconstruction (11, 12)
- Recognition (14)

Specifics of the course

- Lectures on Mondays (12 lectures)
- Exercises on Fridays (12 sessions, 1st on Sep. 14)
  - The solutions to weekly homework assignments should be returned before the session
  - The solutions are presented in the session
- Required: non-zero points from at least 8 weekly assignment rounds (out of 12)
  - Even if you are not able to solve everything, return something if you try; good try will be rewarded!
  - Completing more than minimum will give additional points to the exam
- Exam on December 14 (and another in February)

Guidance for homework

- Guidance session every Thursday at 14:15-16:00
- Teachers are available to give instructions for solving the homework
- Try first by yourself, but ask help if you are stuck

Summary of requirements

- Get more than 0 points from at least 8 exercise rounds
- Pass the exam
- Note about exercise points:
  - Try to do more than the minimum
  - You will learn more and the extra points are also taken into account in grading
  - However, solving all homework tasks is not required for a good grade if you do well in the exam (exam points have larger weight than homework points)

Questions / Feedback

- Questions about arrangements?

Questions / Feedback

- Questions about arrangements?
- Feedback always welcome!
- What is your background in related topics:
  - Computer Graphics? (e.g. CS-C3100)
  - Digital Signal Processing? (e.g. ELEC-C5230)
  - Machine Learning? (e.g. CS-E3210)
  - Deep Learning? (e.g. CS-E4890)

Lecture 1b: Image formation

- Geometric transformations
  - Perspective camera: 3D to 2D projection
  - Homogeneous coordinates
  - Transformations in 2D and 3D
- Photometric image formation
- The digital camera

Lecture 1b: Image formation

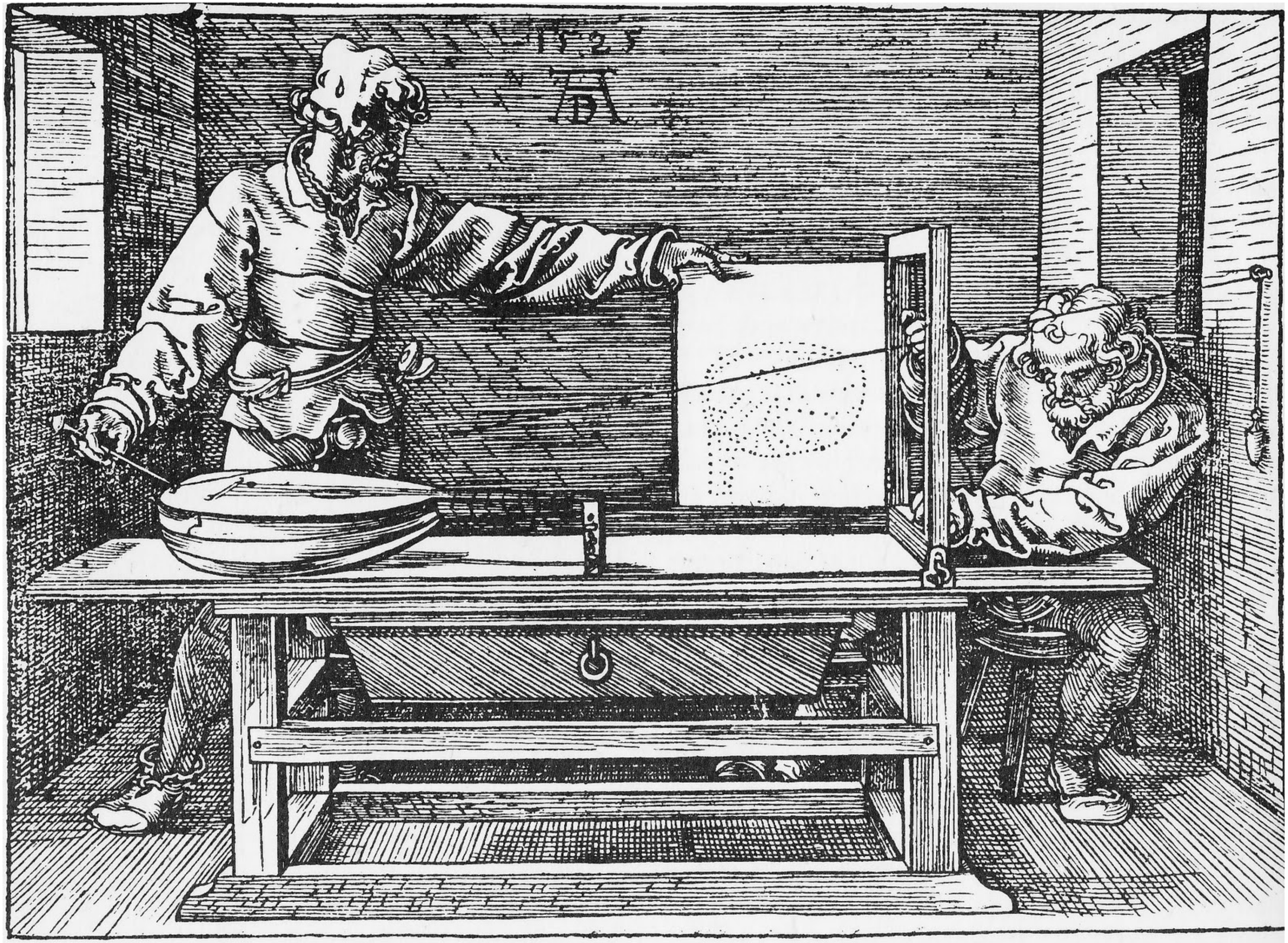
- Geometric transformations
  - Perspective camera: 3D to 2D projection
  - Homogeneous coordinates
  - Transformations in 2D and 3D
- Photometric image formation
- The digital camera

Important information for understanding the basic building blocks of modern vision systems!

Relevant reading

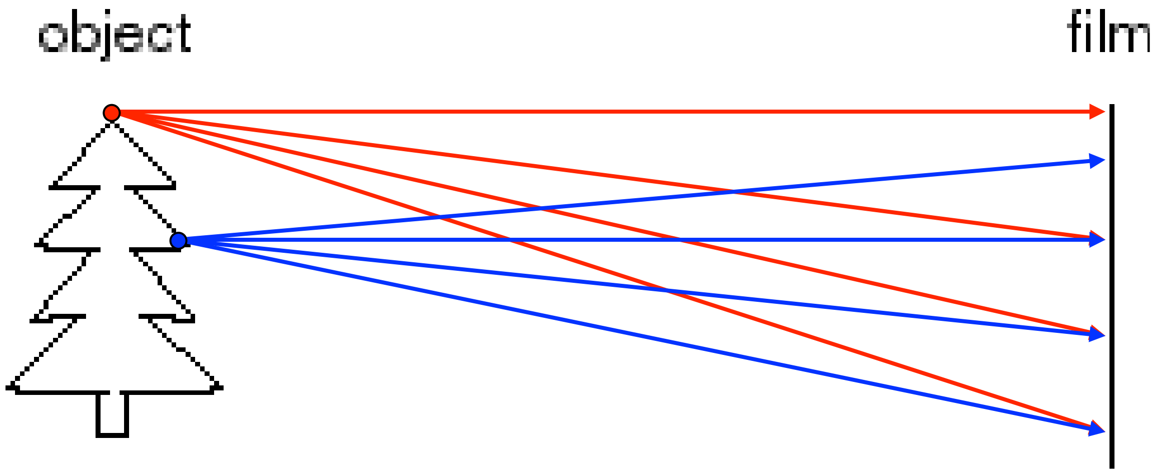
- Chapters 2, 3 and 6 in [Hartley & Zisserman]
  - Comprehensive presentation of the core content of this lecture
  - Useful for understanding latter parts of the course
- Chapter 2 in [Szeliski]
  - Gives a broader overview of image formation

Perspective projection



Albrecht Dürer, *Mechanical creation of a perspective image*, 1525

Let's design a camera

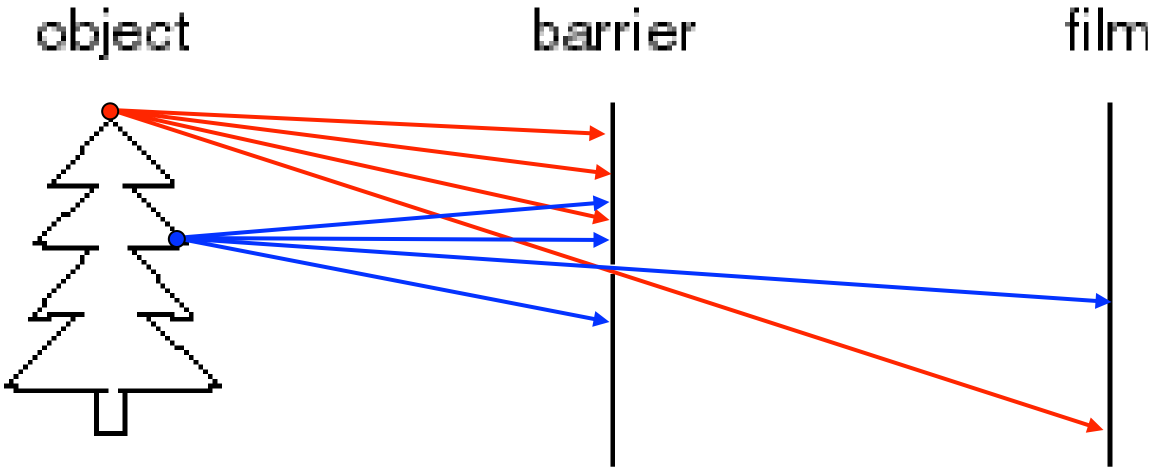


Idea 1: put a piece of film in front of an object

Do we get a reasonable image?

Slide by Steve Seitz

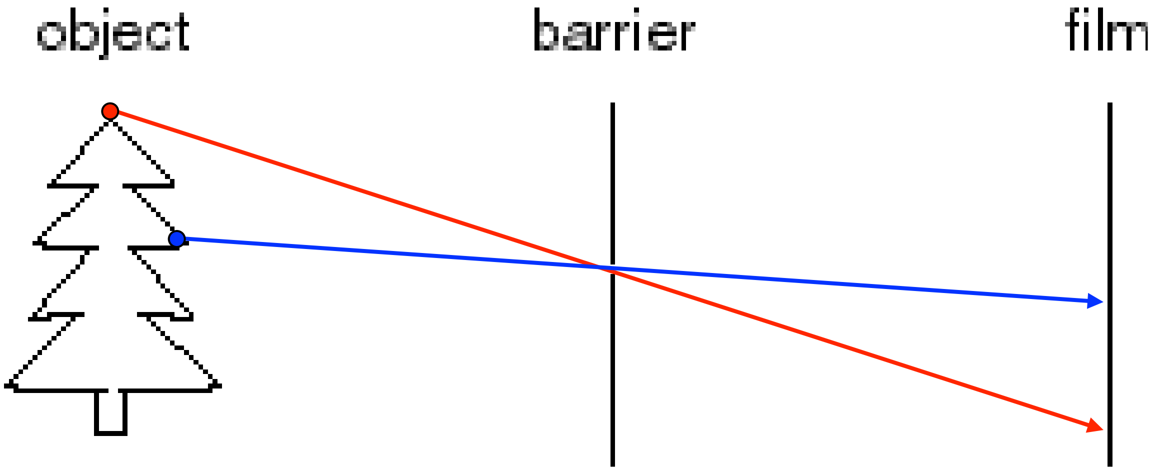
Pinhole camera



Add a barrier to block off most of the rays

Slide by Steve Seitz

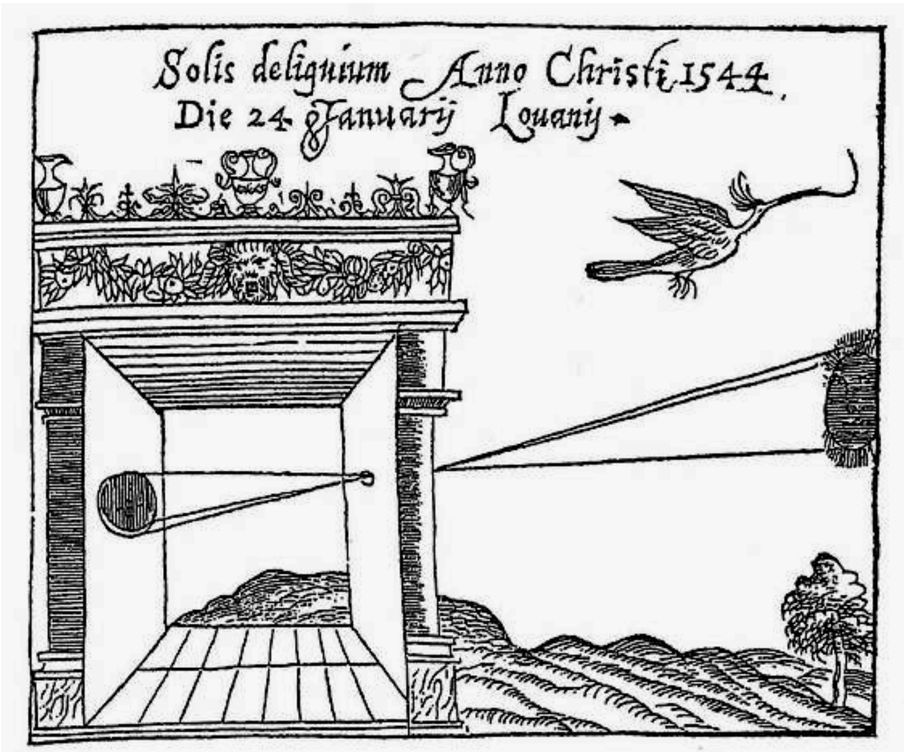
Pinhole camera



- Captures **pencil of rays** – all rays through a single point: **aperture, center of projection, optical center, focal point, camera center**
- The image is formed on the **image plane**

Slide by Steve Seitz

Camera obscura



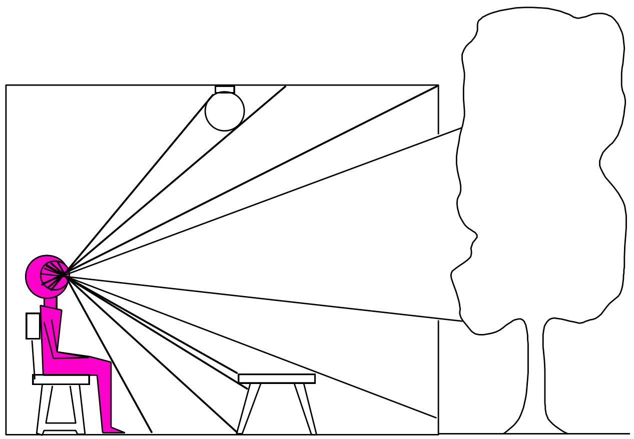
Gemma Frisius, 1558

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

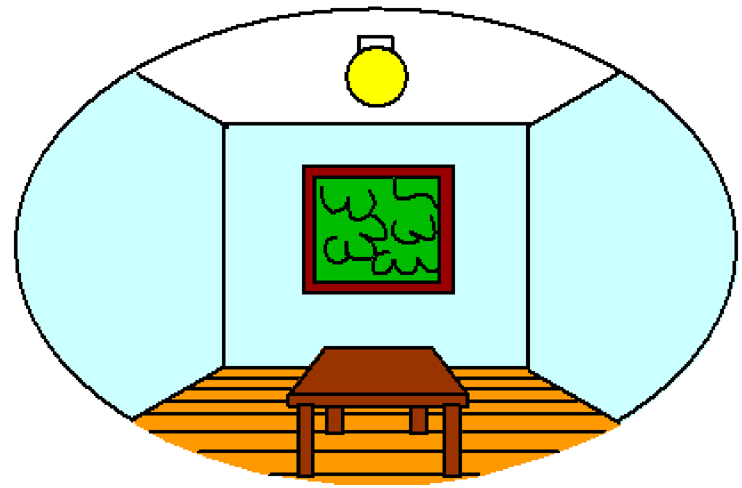
Source: A. Efros

Dimensionality reduction: from 3D to 2D

3D world *2D image*



Point of observation



What properties of the world are preserved?

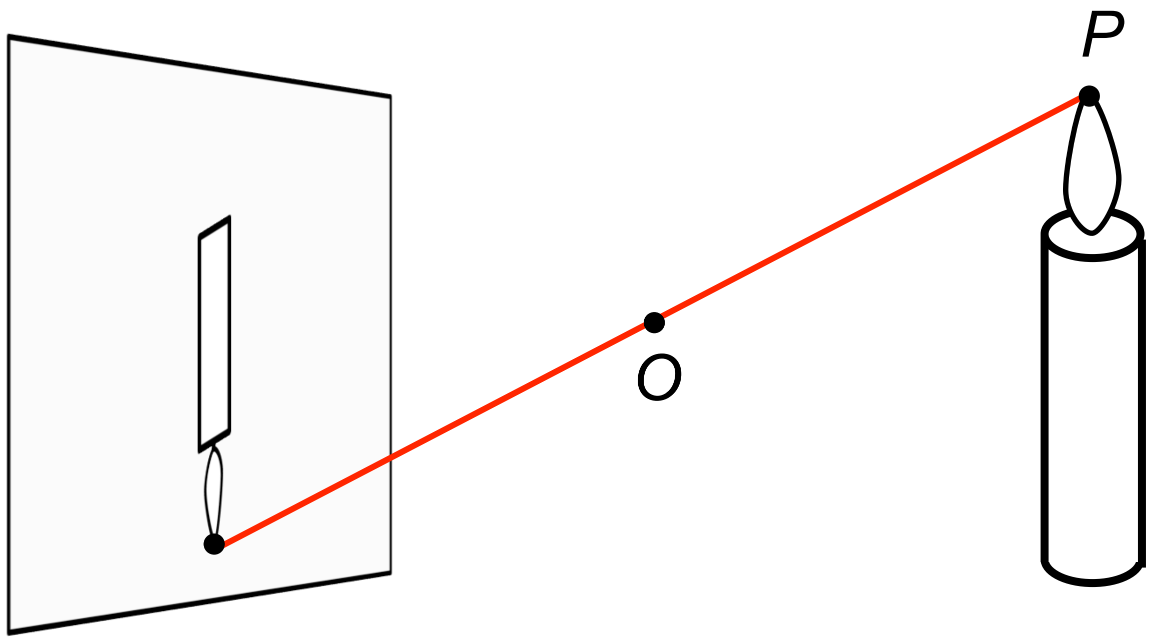
- Straight lines, incidence

What properties are not preserved?

- Angles, lengths

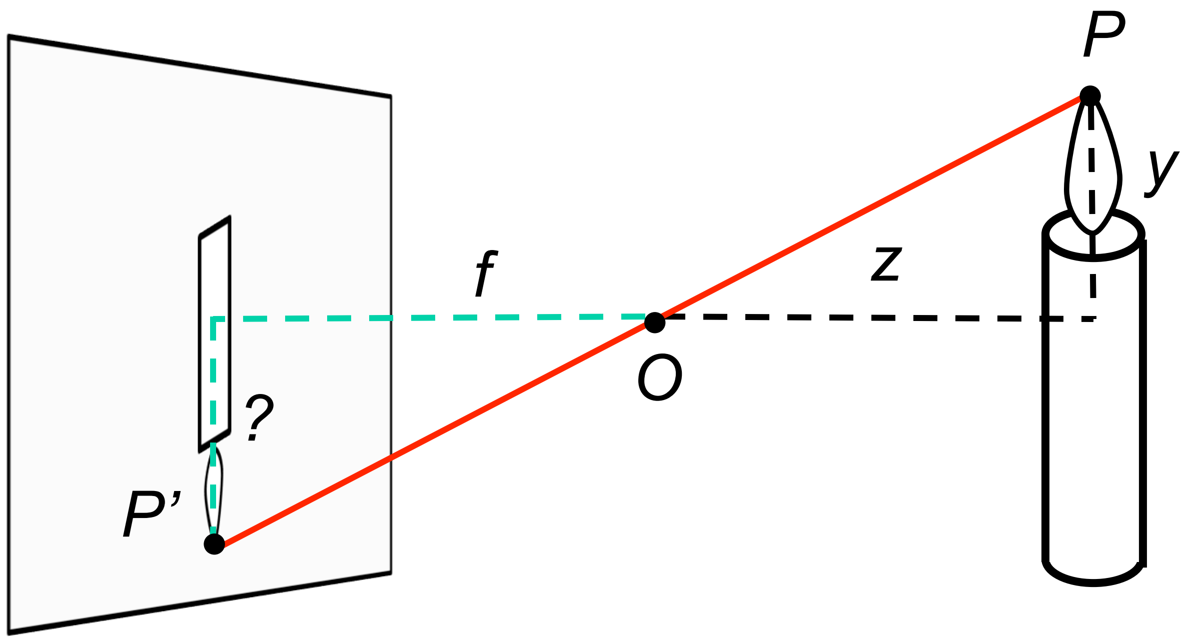
Slide by A. Efros

Modeling projection



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

Modeling projection



The coordinate system

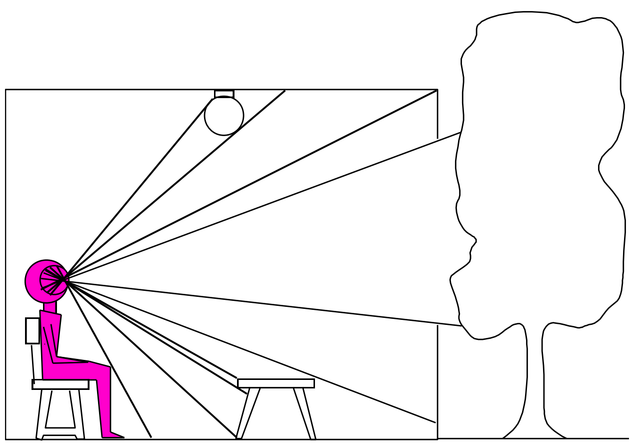
- The optical center (***O***) is at the origin
- The image plane is parallel to xy-plane or perpendicular to the z-axis, which is the *optical axis*

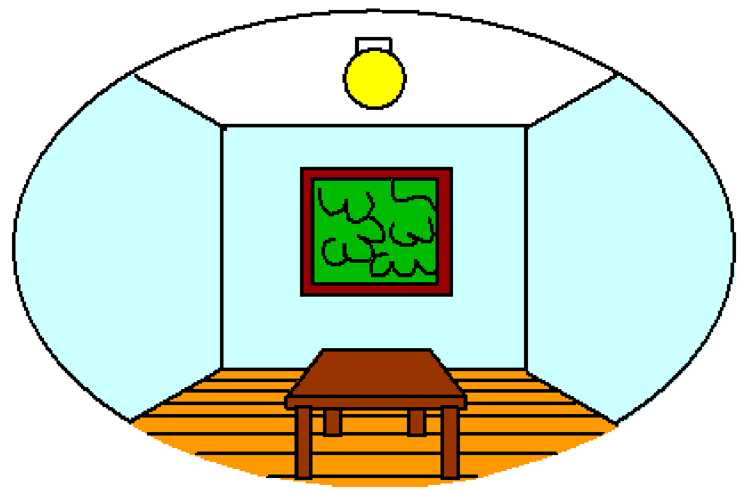
Projection equations

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

Fronto-parallel planes

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

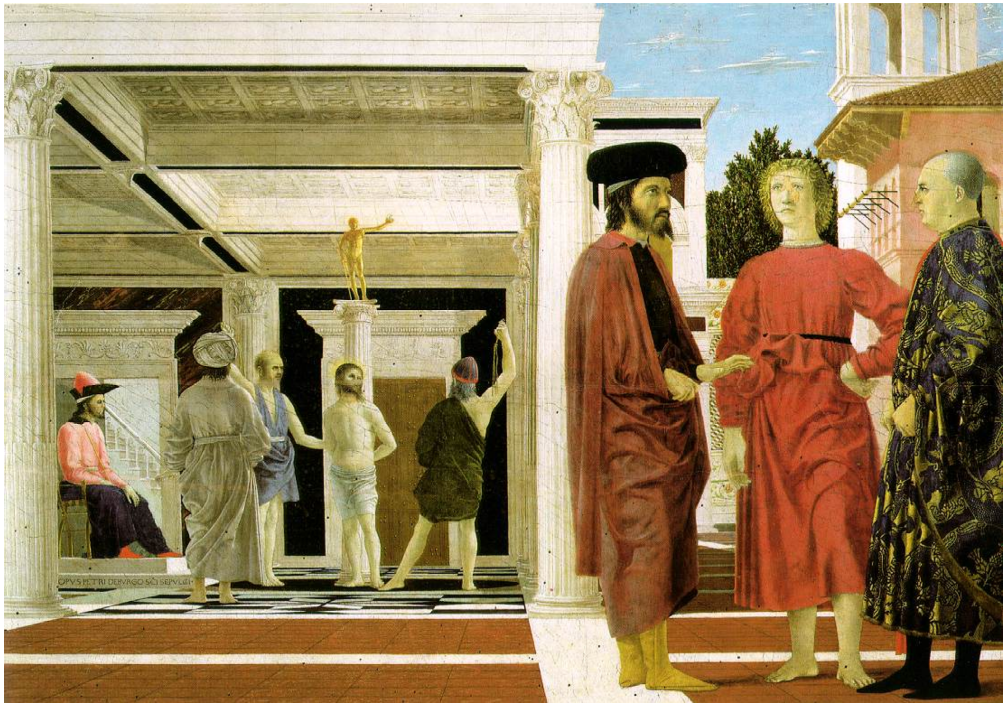




$$(x, y, z) \rightarrow (f\frac{x}{z}, f\frac{y}{z})$$

Fronto-parallel planes

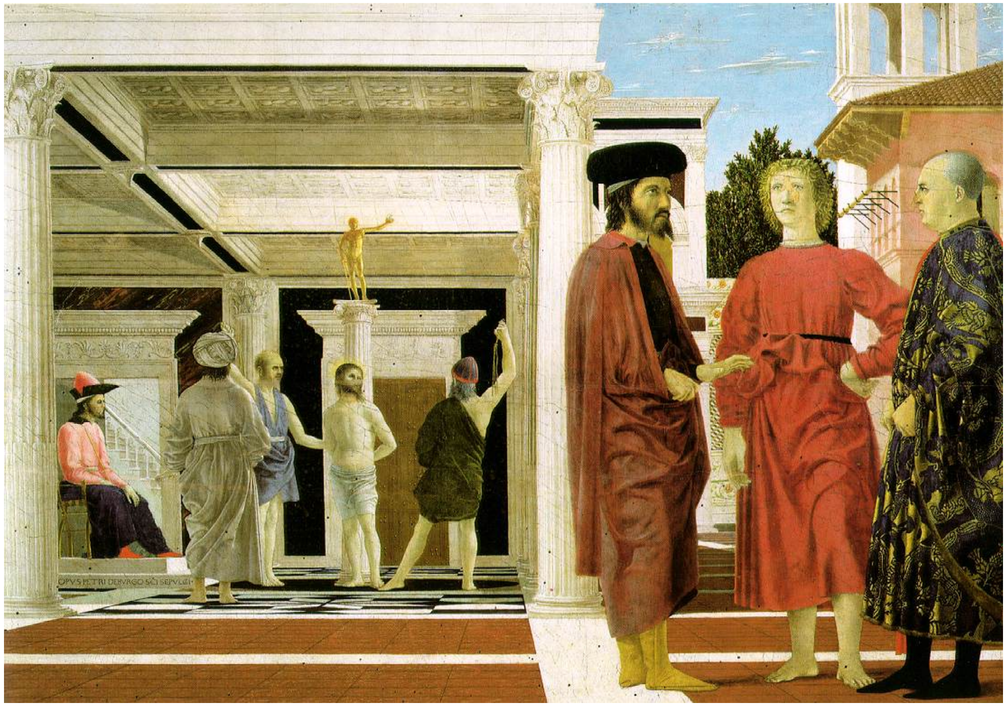
- What happens to the projection of a pattern on a plane parallel to the image plane?
  - All points on that plane are at a fixed *depth z*
  - The pattern gets scaled by a factor of *f* / *z*, but angles and ratios of lengths/areas are preserved



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

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

What about non-fronto-parallel planes?



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

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

Slide source: Seitz

Projection can be tricky…

Slide source: Seitz

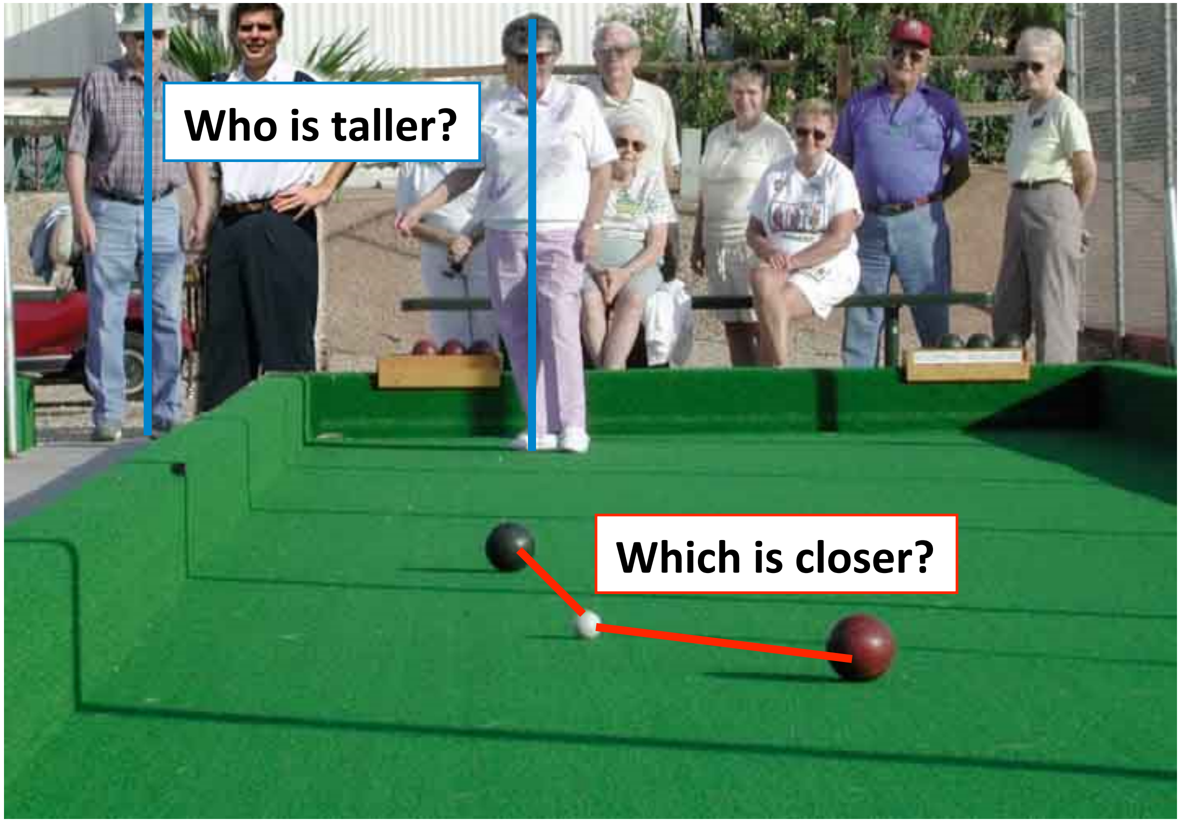
Projection can be tricky…

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

Projective Geometry

What is lost?

- Length



Length is not preserved

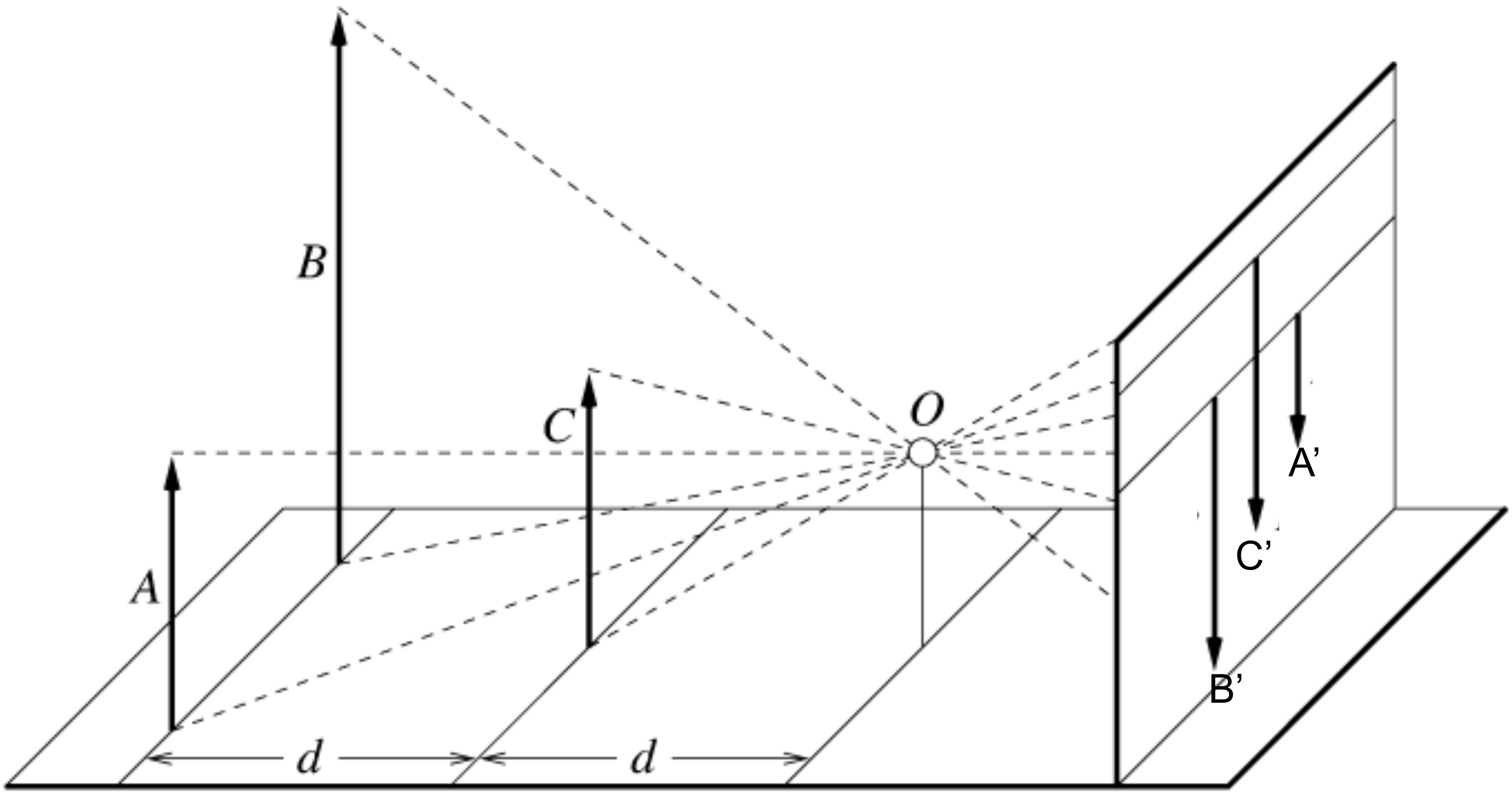
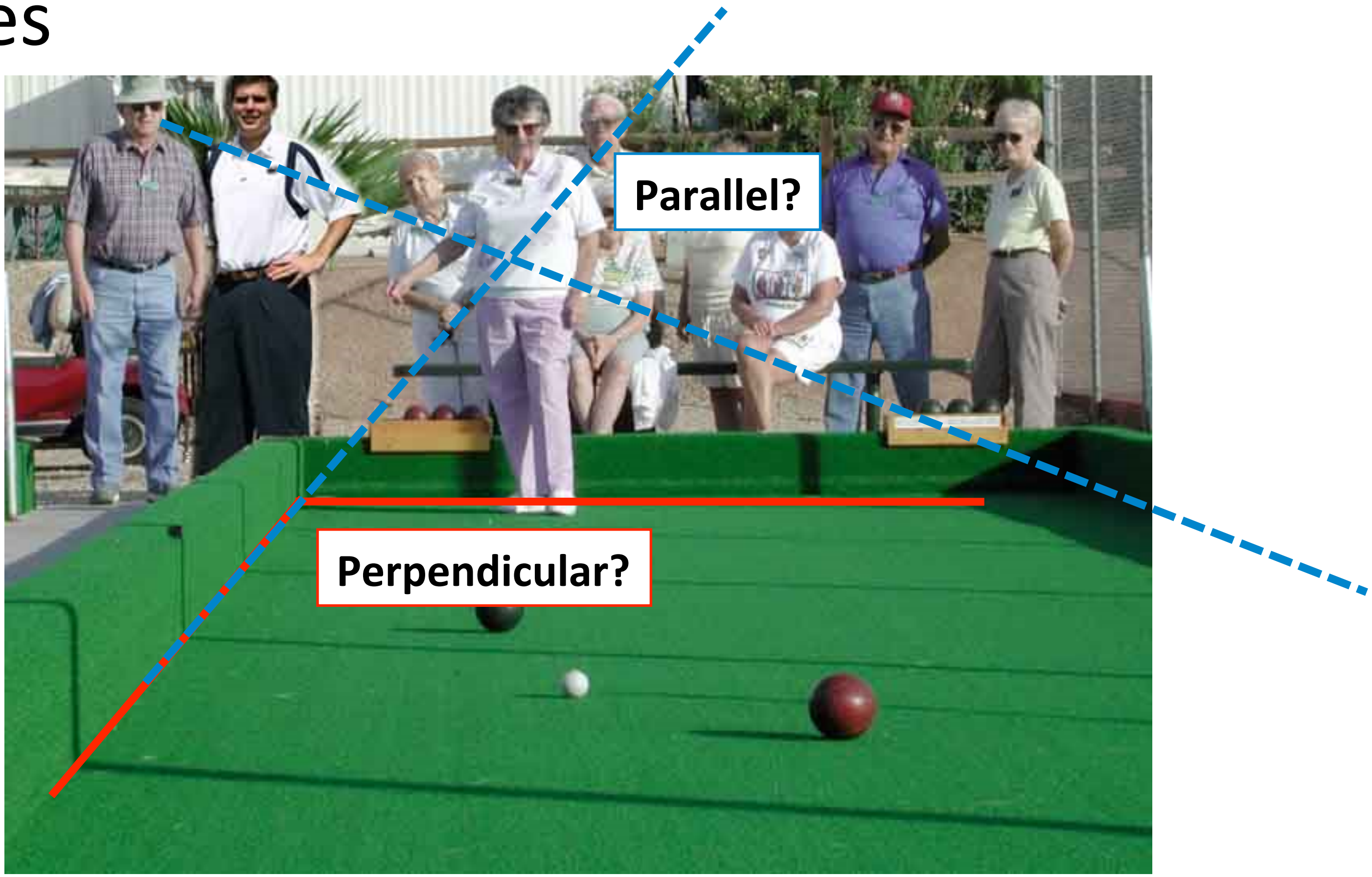


Figure by David Forsyth

Projective Geometry

What is lost?

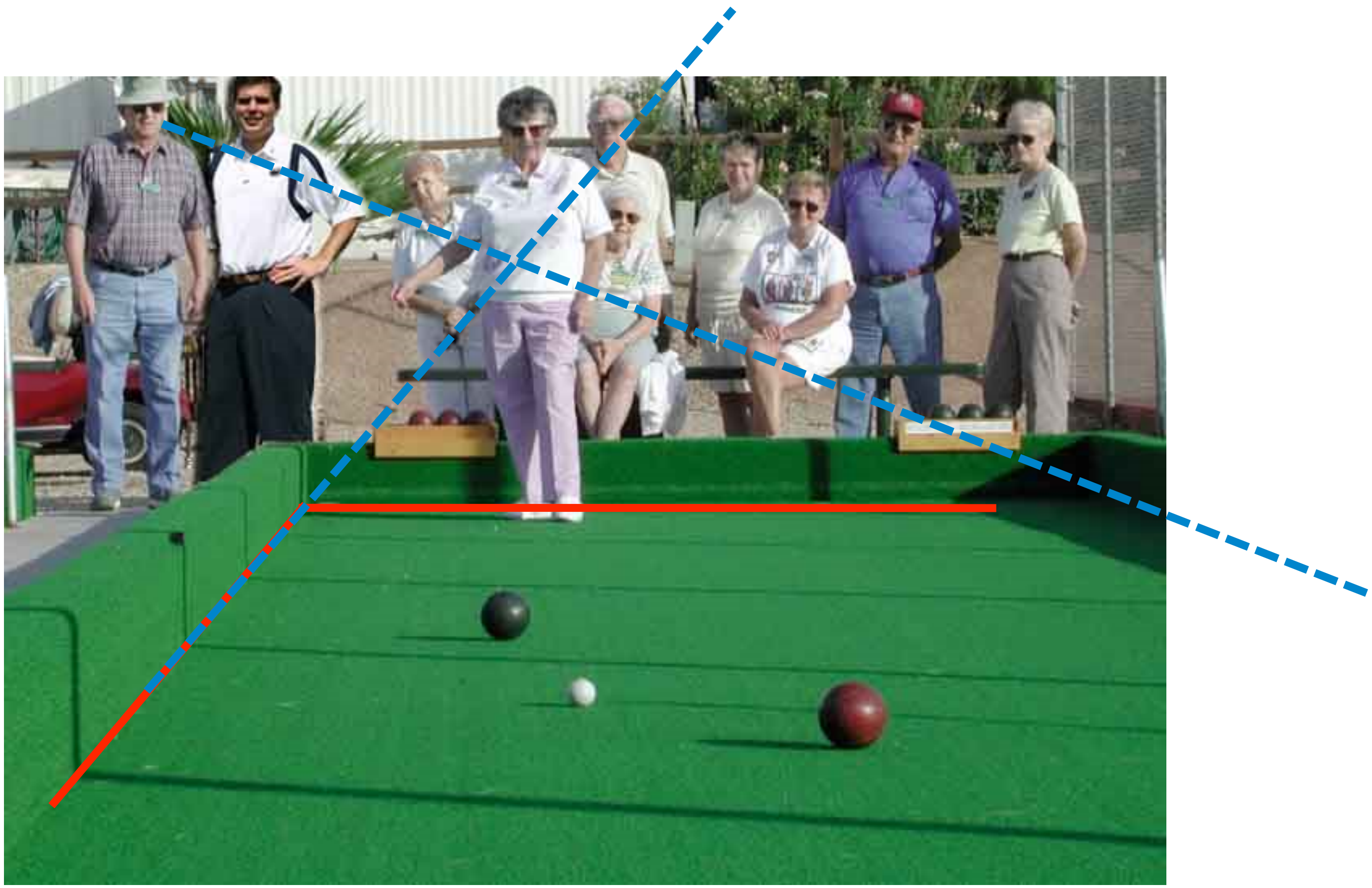
- Length
- Angles



Projective Geometry

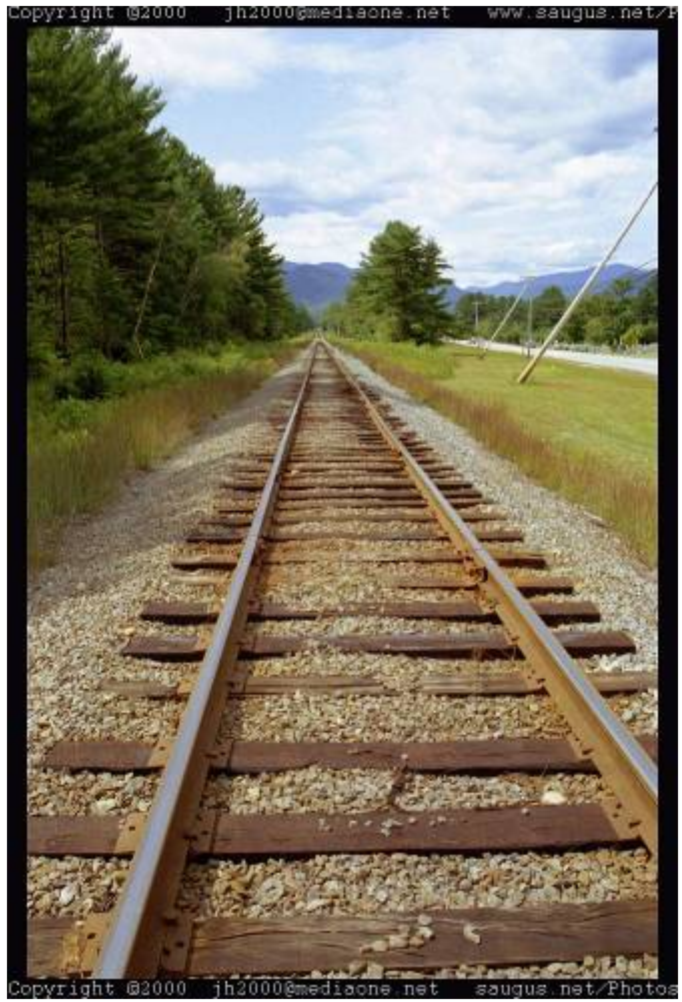
What is preserved?

- Straight lines are still straight

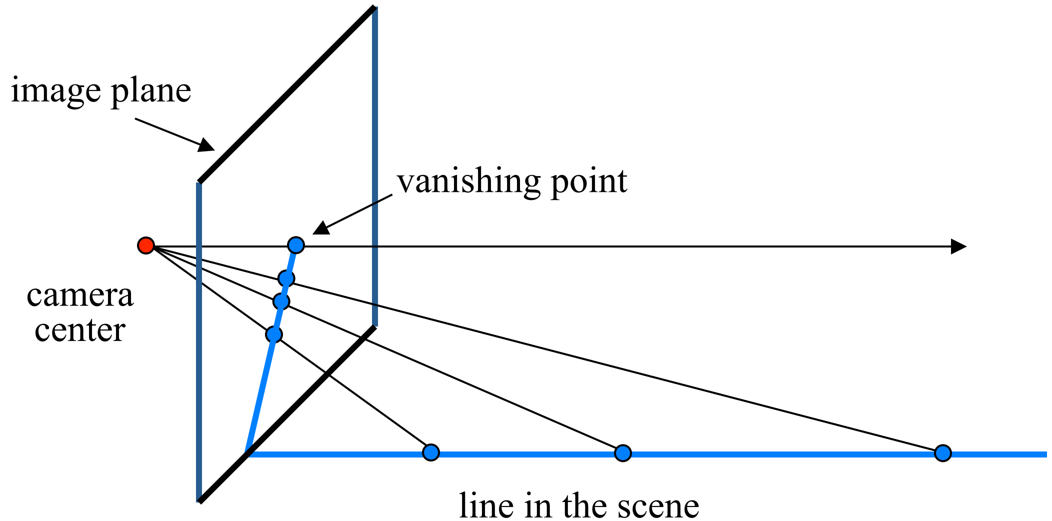


Vanishing points and lines

Parallel lines in the world intersect in the image at a "vanishing point"

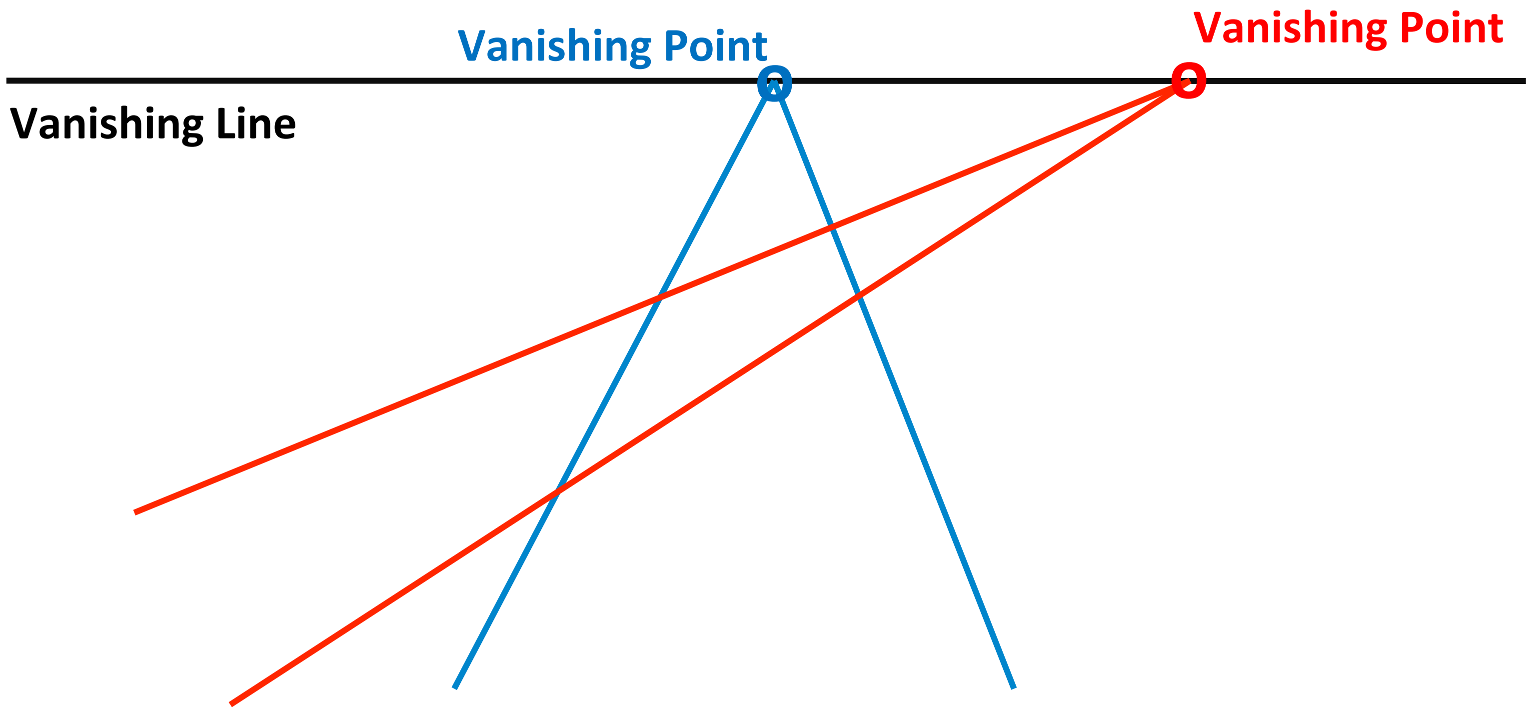


Constructing the vanishing point of a line



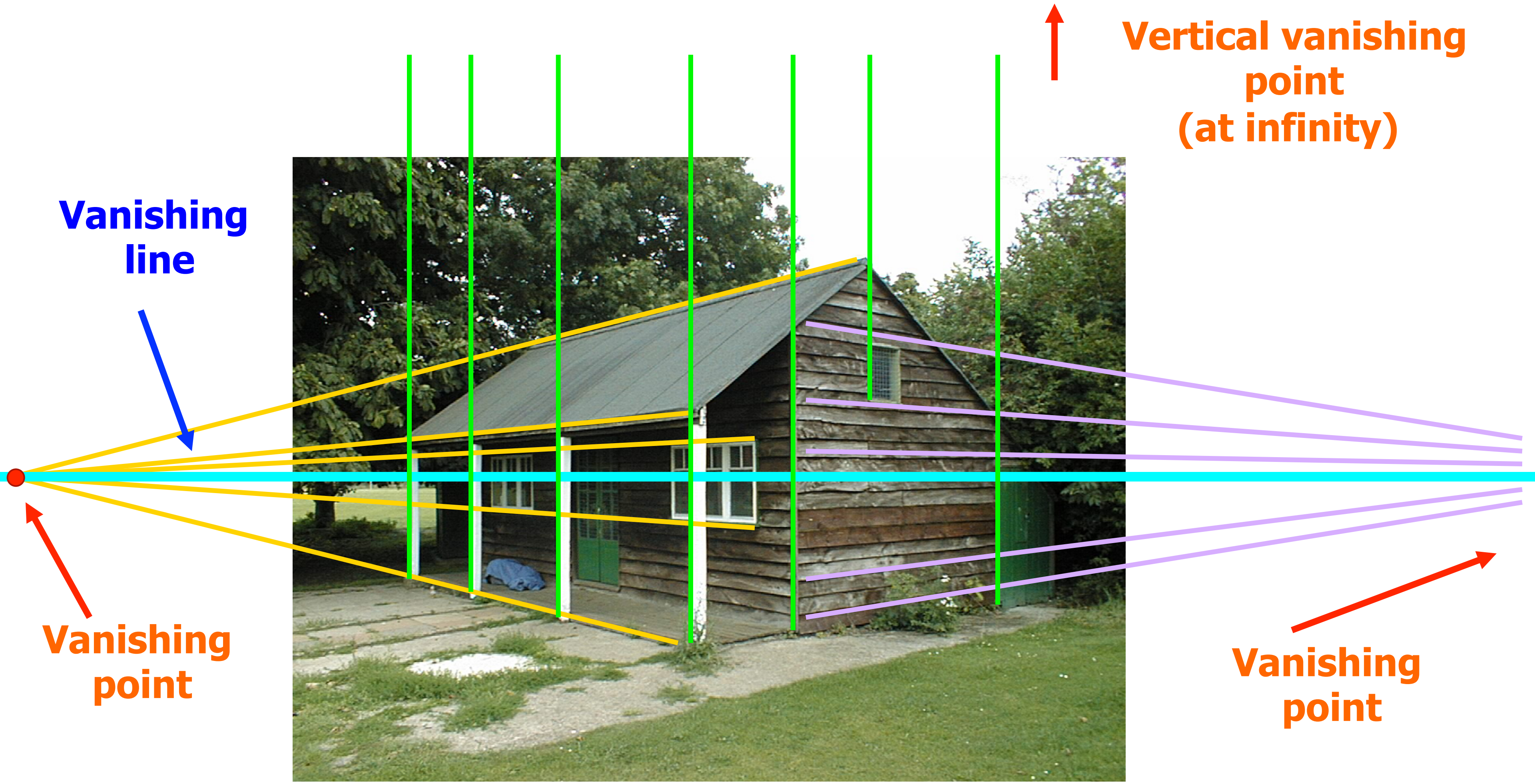
Slide by Steve Seitz

Vanishing points and lines



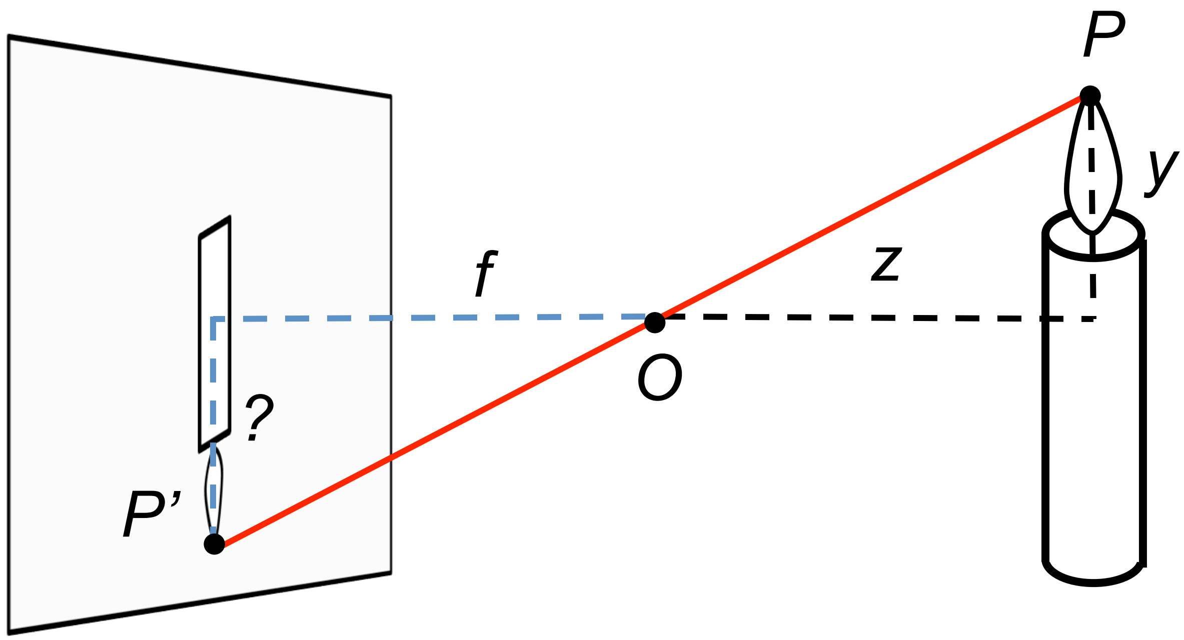
- The projections of parallel 3D lines intersect at a **vanishing point**
- The projection of parallel 3D planes intersect at a **vanishing line**
- If a set of parallel 3D lines are also parallel to a particular plane, their vanishing point will lie on the vanishing line of the plane
- Vanishing point <-> 3D direction of a line
- Vanishing line <-> 3D orientation of a surface

Vanishing points and lines



Slide from Efros, Photo from Criminisi

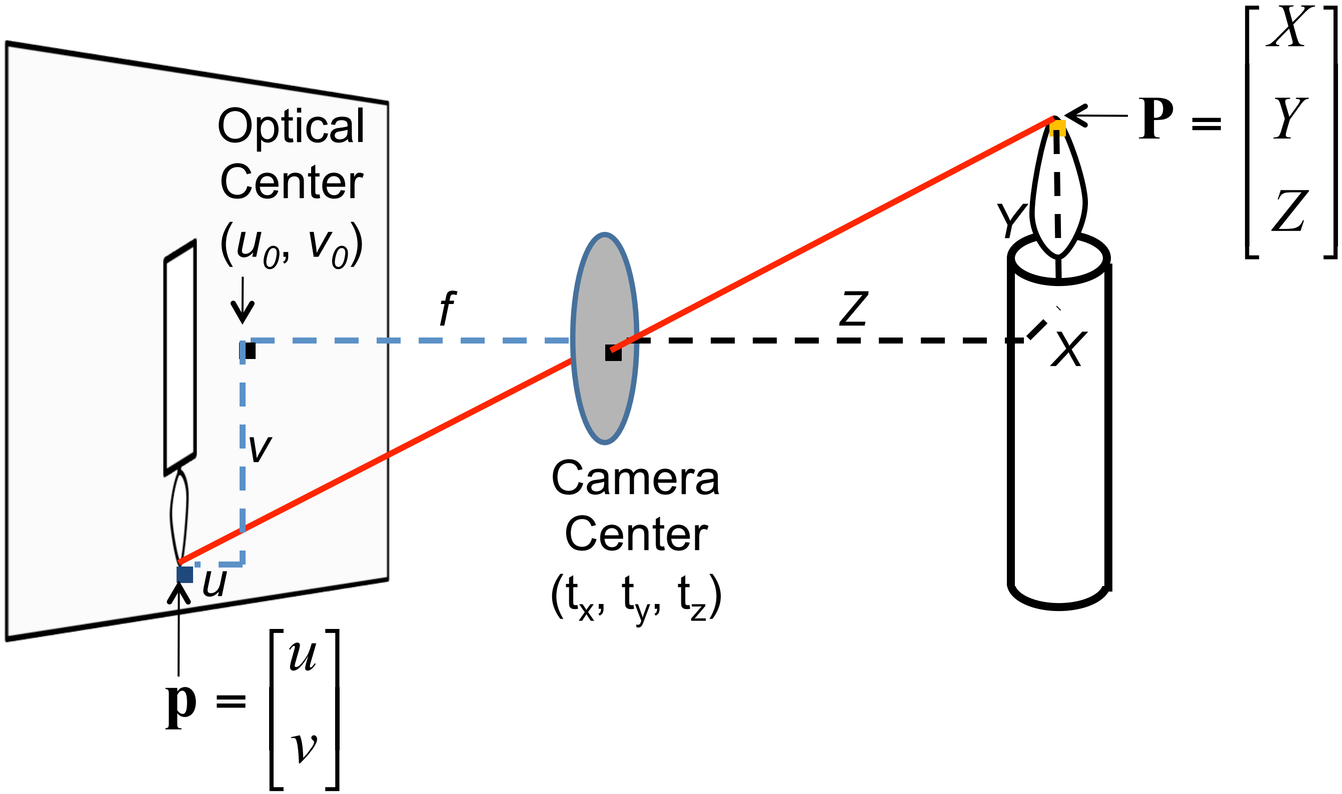
Modeling projection



Projection equations

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

Projection: world coordinates→image coordinates



Homogeneous coordinates

Conversion

Converting to *homogeneous* coordinates

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

homogeneous image coordinates

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

homogeneous scene coordinates

Converting *from* homogeneous coordinates

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

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

Homogeneous coordinates

Invariant to scaling

$$k\begin{bmatrix} x \\ y \\ w \end{bmatrix} = \begin{bmatrix} kx \\ ky \\ kw \end{bmatrix} \Rightarrow \begin{bmatrix} \frac{kx}{kw} \\ \frac{ky}{kw} \end{bmatrix} = \begin{bmatrix} \frac{x}{w} \\ \frac{y}{w} \end{bmatrix}$$

Homogeneous Coordinates

Cartesian Coordinates

Point in Cartesian is ray in Homogeneous

Basic geometry in homogeneous coordinates

- Line equation: ax + by + c = 0

$$line_i = \begin{bmatrix} a_i \\ b_i \\ c_i \end{bmatrix}$$

- Append 1 to pixel coordinate to get homogeneous coordinate

$$p_i = \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix}$$

- Line given by cross product of two points

$$line_{ij} = p_i \times p_j$$

- Intersection of two lines given by cross product of the lines

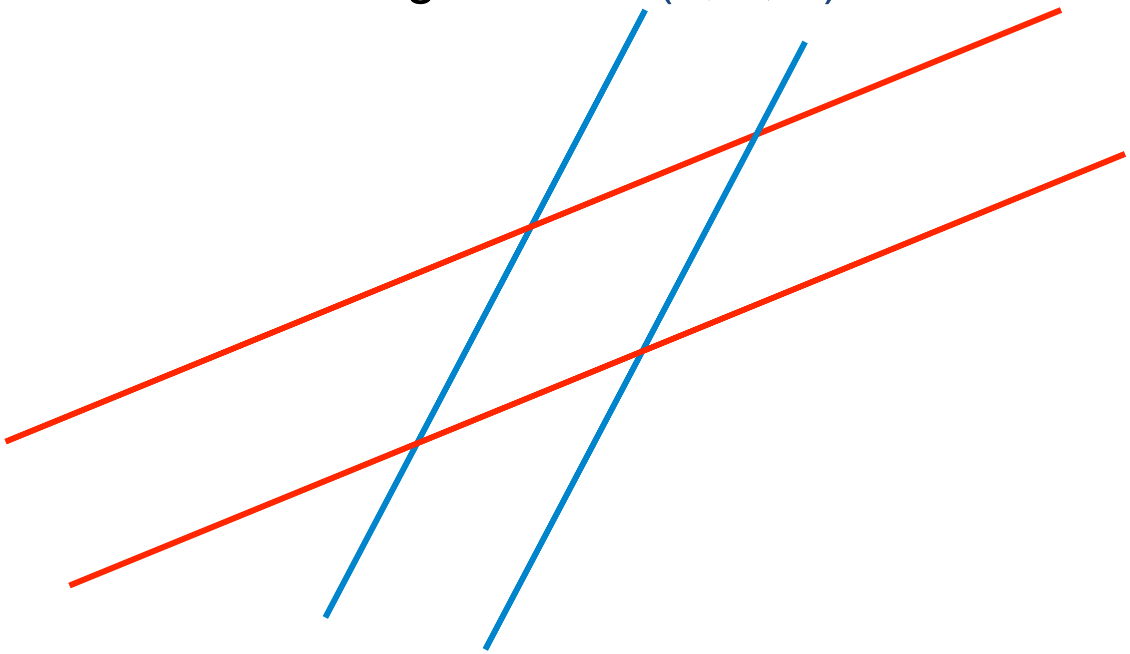
$$q_{ij} = line_i \times line_j$$

Another problem solved by homogeneous coordinates

Intersection of parallel lines

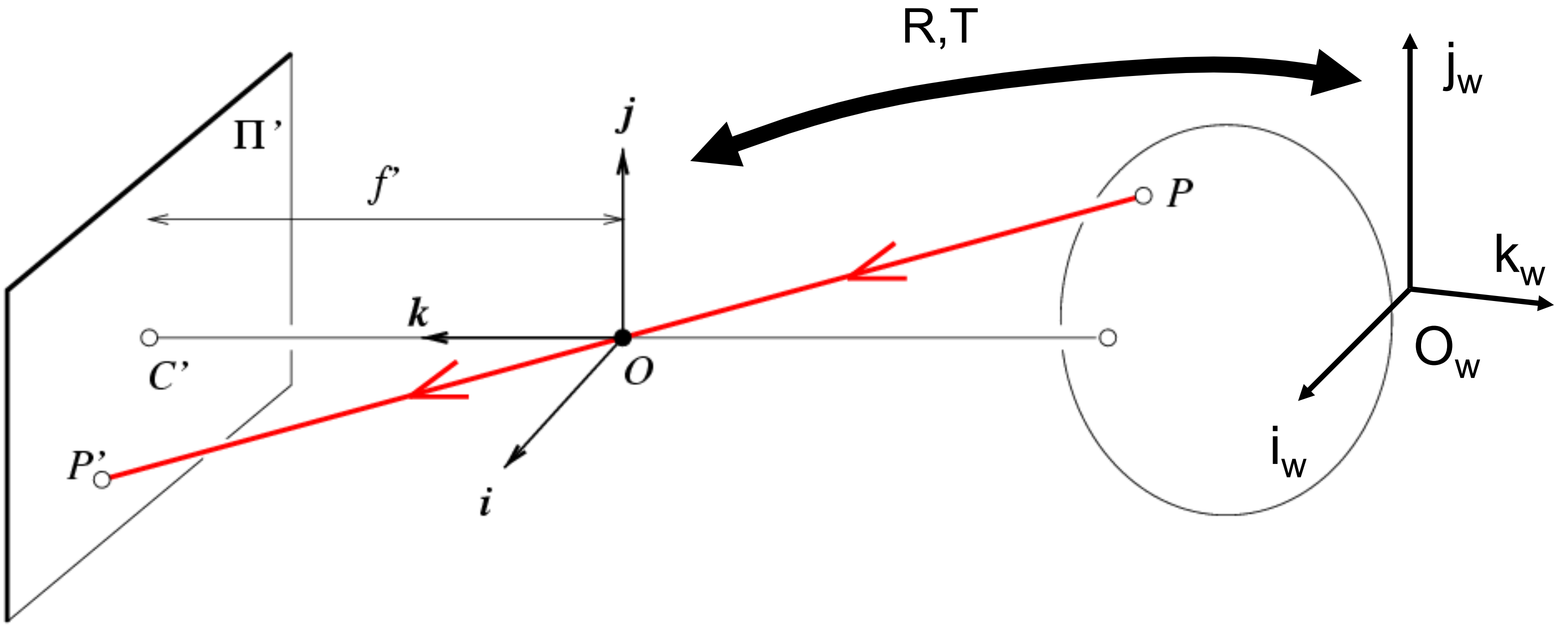
Cartesian: (Inf, Inf)
Homogeneous: (1, 1, 0)

Cartesian: (Inf, Inf)
Homogeneous: (1, 2, 0)



Projection matrix

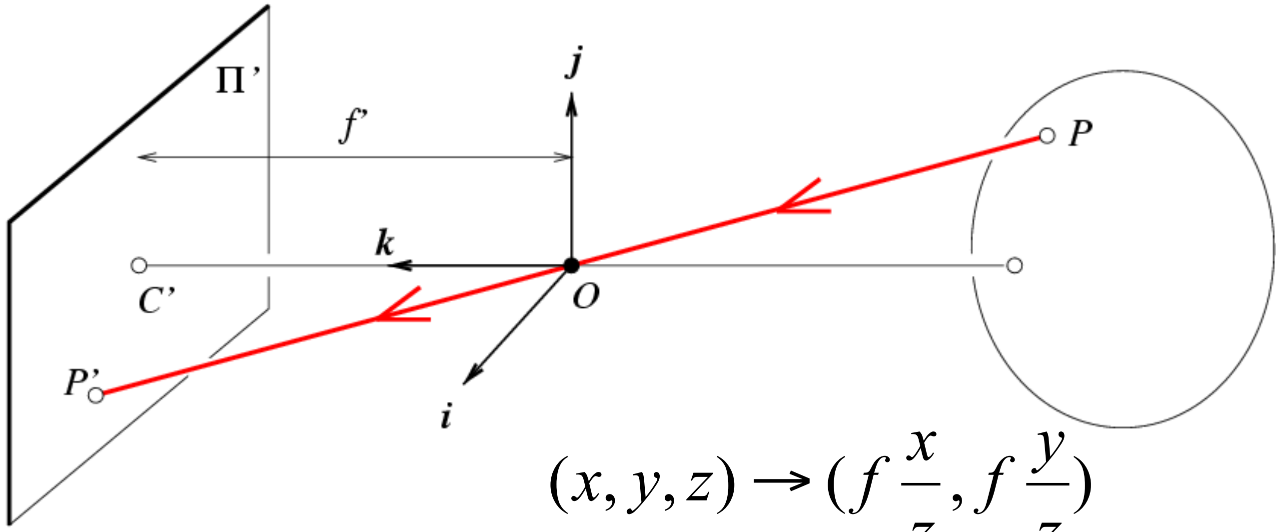
Slide Credit: Savarese



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

x: Image Coordinates: (u,v,1)
K: Intrinsic Matrix (3x3)
R: Rotation (3x3)
t: Translation (3x1)
X: World Coordinates: (X,Y,Z,1)

Projection matrix



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

Intrinsic Assumptions

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

Extrinsic Assumptions

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

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix}u \\ v \\ 1\end{bmatrix} = \begin{bmatrix}\boxed{\begin{matrix}f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1\end{matrix}}_{\mathbf{K}} & \begin{matrix}0 \\ 0 \\ 0\end{matrix}\end{bmatrix}\begin{bmatrix}x \\ y \\ z \\ 1\end{bmatrix}$$

Slide Credit: Savarese

Remove assumption: known optical center

Intrinsic Assumptions
- Unit aspect ratio
- No skew

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

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

Remove assumption: square pixels

Intrinsic Assumptions
- No skew

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

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

Remove assumption: non-skewed pixels

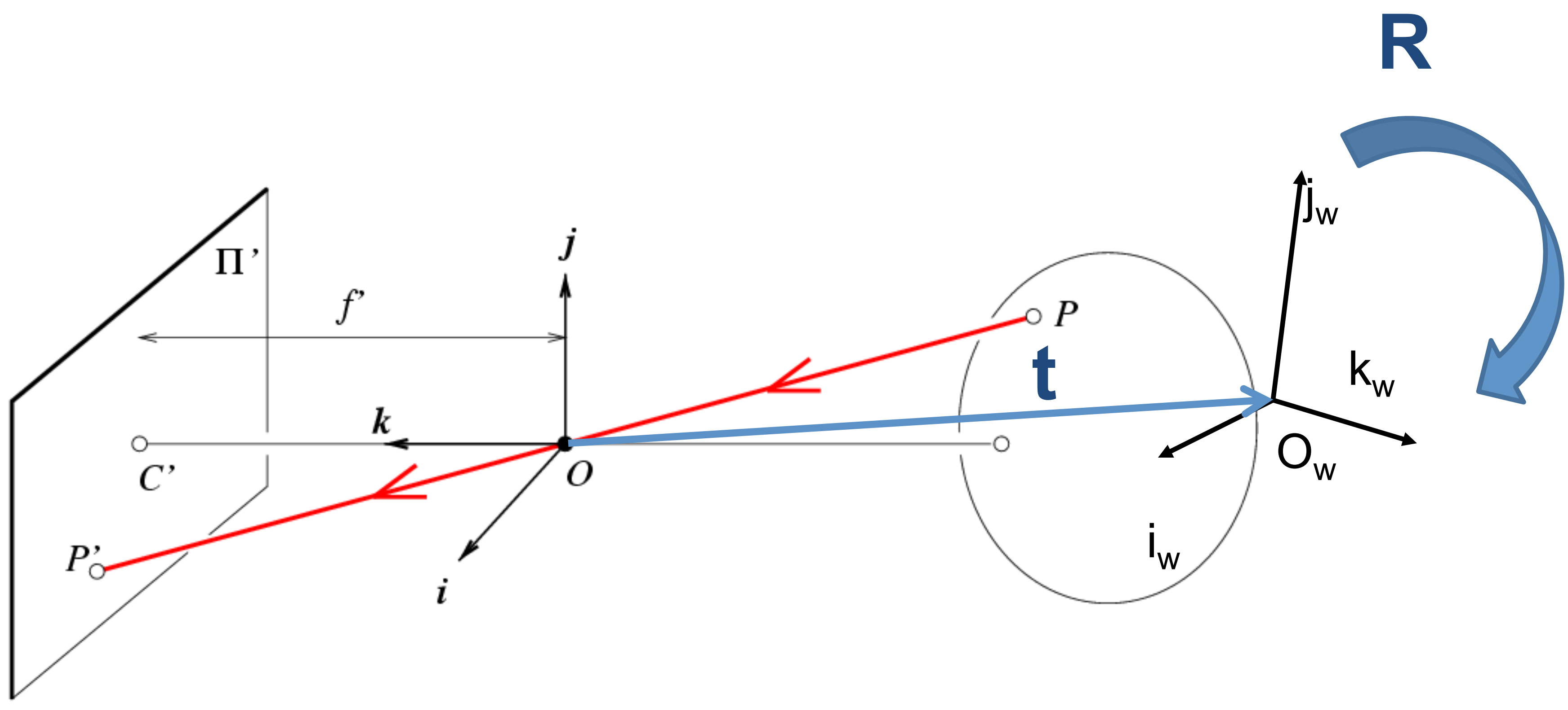
Intrinsic Assumptions

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

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

Note: different books use different notation for parameters

Oriented and Translated Camera



Allow camera translation

Intrinsic Assumptions

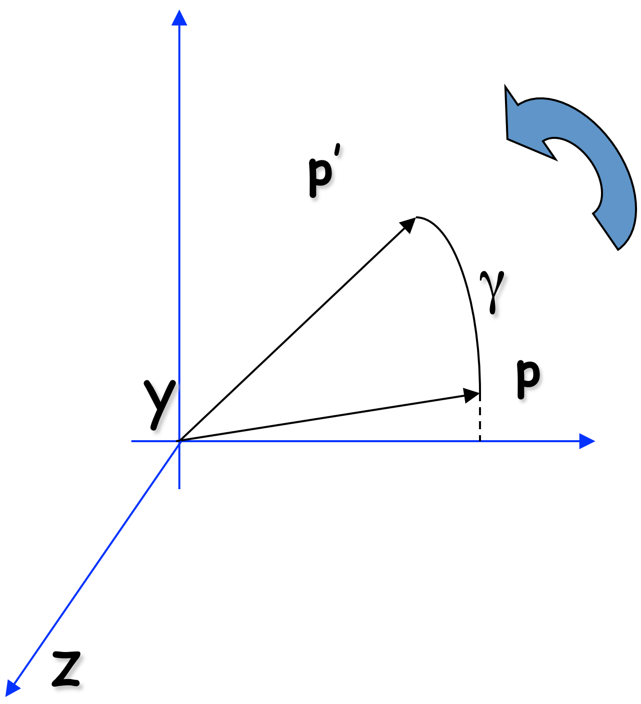
Extrinsic Assumptions
- No rotation

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{t}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}\alpha & 0 & u_0\\0 & \beta & v_0\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & t_x\\0 & 1 & 0 & t_y\\0 & 0 & 1 & t_z\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

Slide Credit: Saverese

3D Rotation of Points

Rotation around the coordinate axes, counter-clockwise:



$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

$$R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Allow camera rotation

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

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

Degrees of freedom

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

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

Degrees of freedom

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

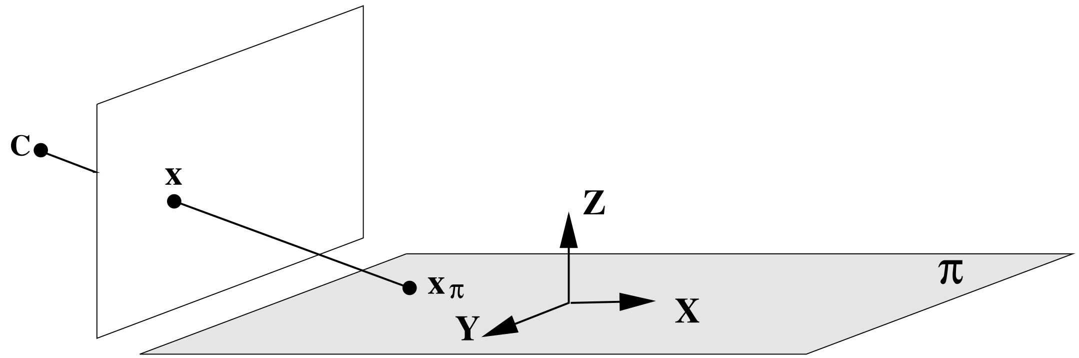
$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \overset{5}{\begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}} \overset{6}{\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}} \begin{bmatrix} x \\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

Vanishing Point = Projection from Infinity

$$\mathbf{p} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 0 \end{bmatrix} \Rightarrow \mathbf{p} = \mathbf{KR}\begin{bmatrix} x \\ y \\ z \end{bmatrix} \Rightarrow \mathbf{p} = \mathbf{K}\begin{bmatrix} x_R \\ y_R \\ z_R \end{bmatrix}$$

$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_R \\ y_R \\ z_R \end{bmatrix} \Rightarrow \begin{aligned} u &= \frac{f x_R}{z_R} + u_0 \\ v &= \frac{f y_R}{z_R} + v_0 \end{aligned}$$

Plane projective transformations



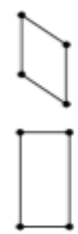
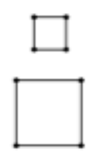
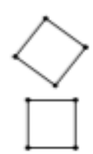
Choose the world coordinate system such that the plane of the points has zero Z coordinate. Then the 3×4 matrix $\mathtt{P}$ reduces to

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ 0 \\ 1 \end{pmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{14} \\ p_{21} & p_{22} & p_{24} \\ p_{31} & p_{32} & p_{34} \end{bmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ 1 \end{pmatrix}$$

which is a 3×3 matrix representing a general plane to plane projective transformation.

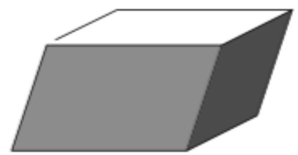
Slide by Hartley & Zisserman

Coordinate transformations in 2D

Group	Matrix	Distortion	Invariant properties
Projective 8 dof	$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$		Concurrency, collinearity, **order of contact**: intersection (1 pt contact); tangency (2 pt contact); inflections (3 pt contact with line); tangent discontinuities and cusps. cross ratio (ratio of ratio of lengths).
Affine 6 dof	$\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		Parallelism, ratio of areas, ratio of lengths on collinear or parallel lines (e.g. midpoints), linear combinations of vectors (e.g. centroids). The line at infinity, $\mathbf{l}_\infty$.
Similarity 4 dof	$\begin{bmatrix} sr_{11} & sr_{12} & t_x \\ sr_{21} & sr_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		Ratio of lengths, angle. The circular points, **I**, **J** (see section 2.7.3).
Euclidean 3 dof	$\begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		Length, area

Slide from Hartley & Zisserman

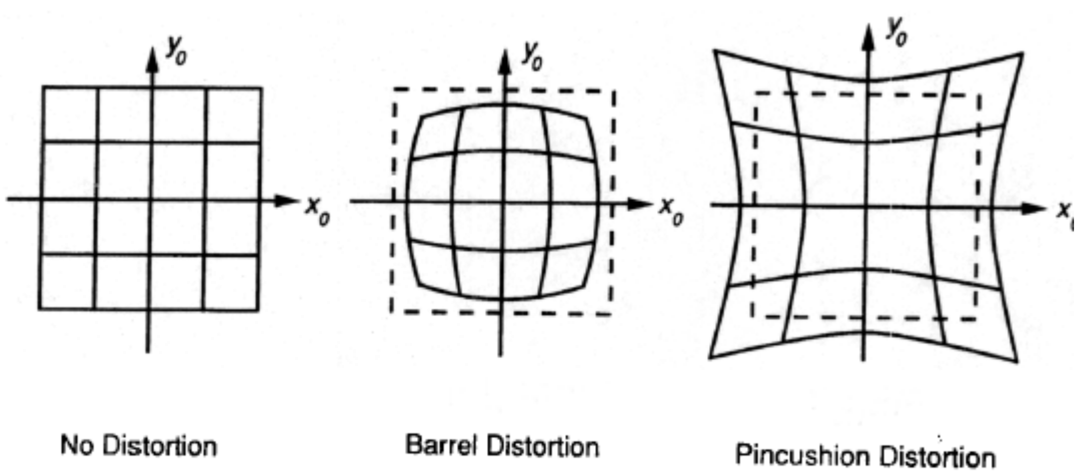
Coordinate transformations in 3D

Group	Matrix	Distortion	Invariant properties
Projective 15 dof	$\begin{bmatrix} \mathtt{A} & \mathbf{t} \\ \mathbf{v}^\mathsf{T} & v \end{bmatrix}$		Intersection and tangency of surfaces in contact. Sign of Gaussian curvature.
Affine 12 dof	$\begin{bmatrix} \mathtt{A} & \mathbf{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}$		Parallelism of planes, volume ratios, centroids. The plane at infinity, $\boldsymbol{\pi}_\infty$, (see section 3.5).
Similarity 7 dof	$\begin{bmatrix} s\mathtt{R} & \mathbf{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}$		The absolute conic, Ω_∞, (see section 3.6).
Euclidean 6 dof	$\begin{bmatrix} \mathtt{R} & \mathbf{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}$		Volume.

Slide from Hartley & Zisserman

Beyond Pinholes: Radial Distortion

- Common in wide-angle lenses or for special applications (e.g., security)
- Creates non-linear terms in projection
- Usually handled by through solving for non-linear terms and then correcting image

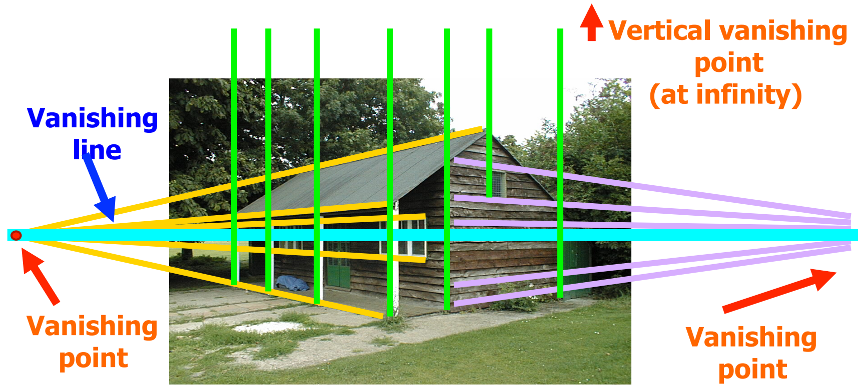


Corrected Barrel Distortion

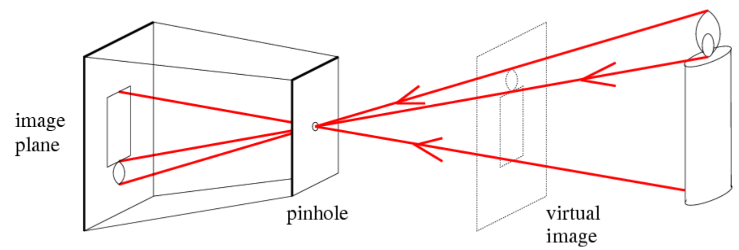
Image from Martin Habbecke

Things to remember

- Vanishing points and vanishing lines



- Pinhole camera model and camera projection matrix



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

- Homogeneous coordinates

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

Lecture 1b: Image formation

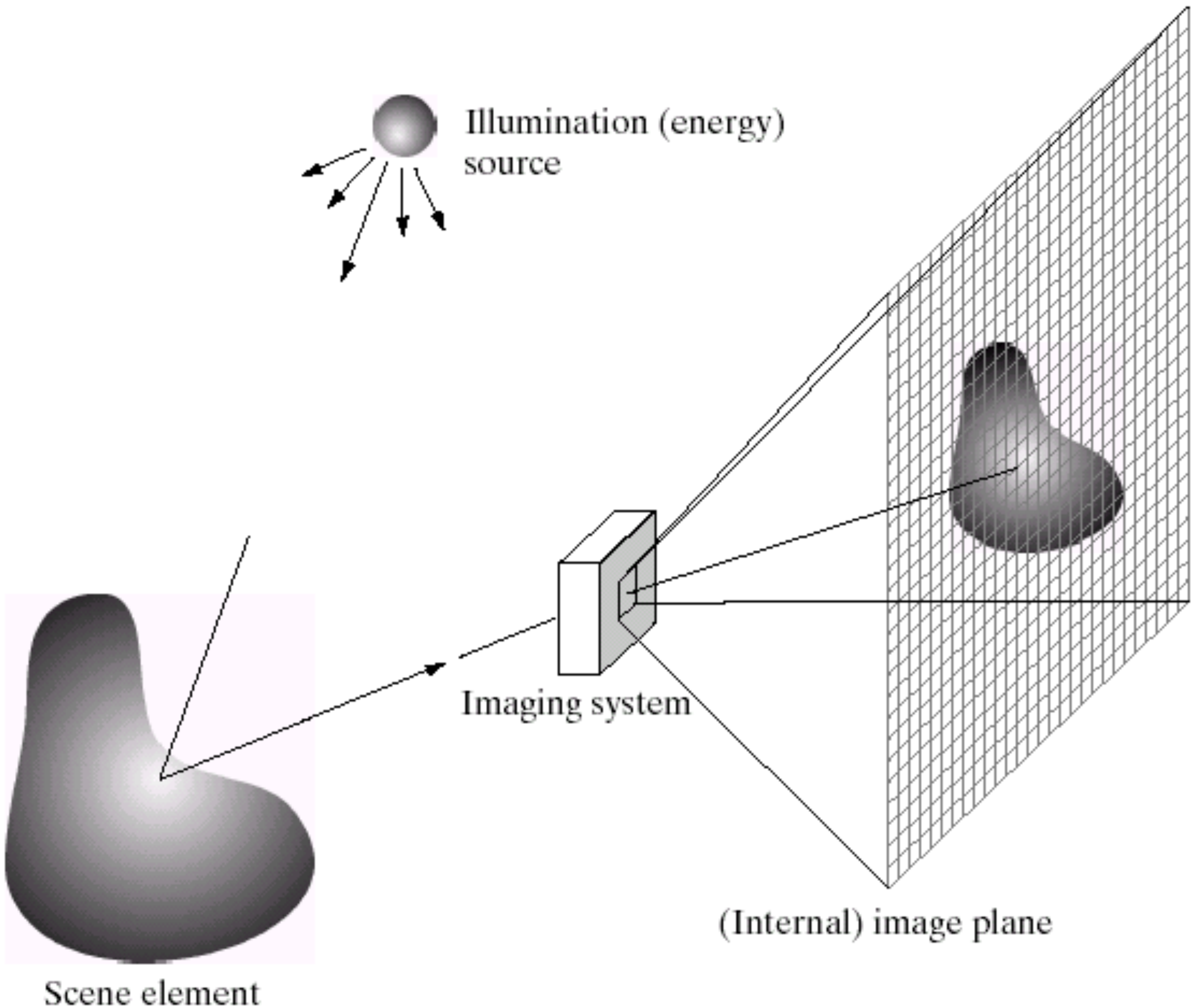
- Geometric transformations
  - Perspective camera: 3D to 2D projection
  - Homogeneous coordinates
  - Transformations in 2D and 3D
- Photometric image formation
- The digital camera

Light and shading

- What determines a pixel's brightness?
  - How is light reflected?
  - How is light measured?

Many of the following slides from Derek Hoiem

How light is recorded



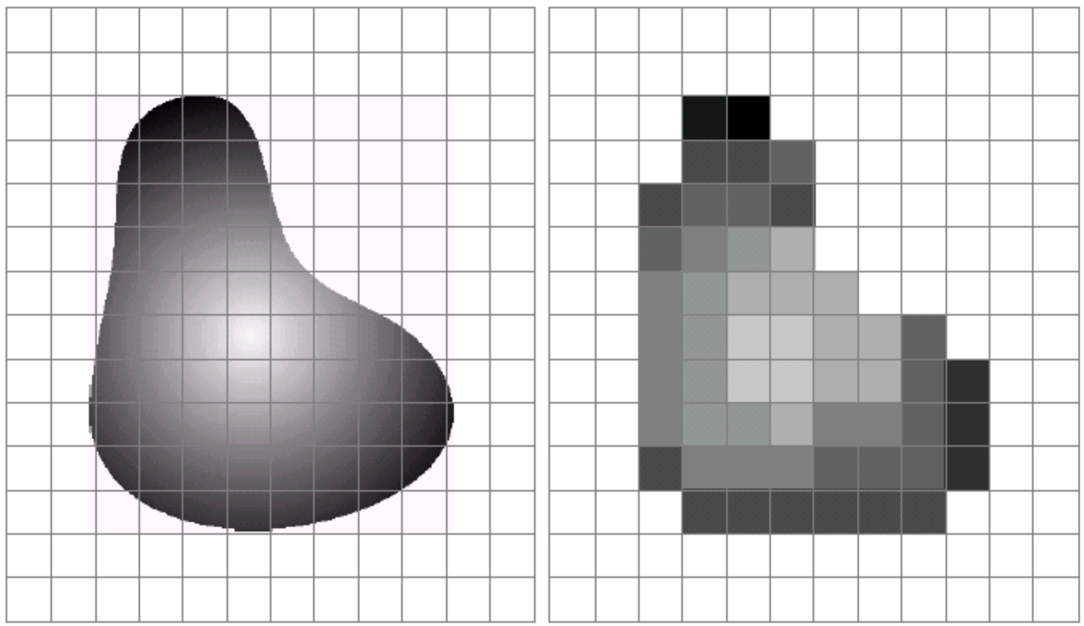
Digital camera

A digital camera replaces film with a sensor array

- Each cell in the array is light-sensitive diode that converts photons to electrons
- Two common types: Charge Coupled Device (CCD) and CMOS
- http://electronics.howstuffworks.com/digital-camera.htm

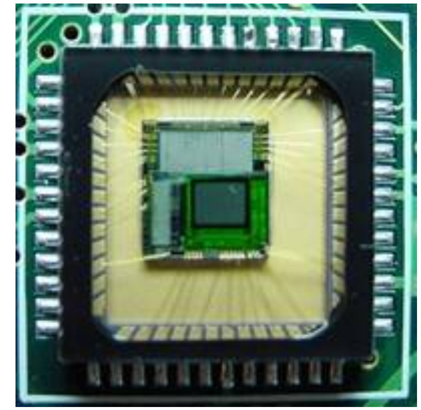
Slide by Steve Seitz

Sensor Array



a b

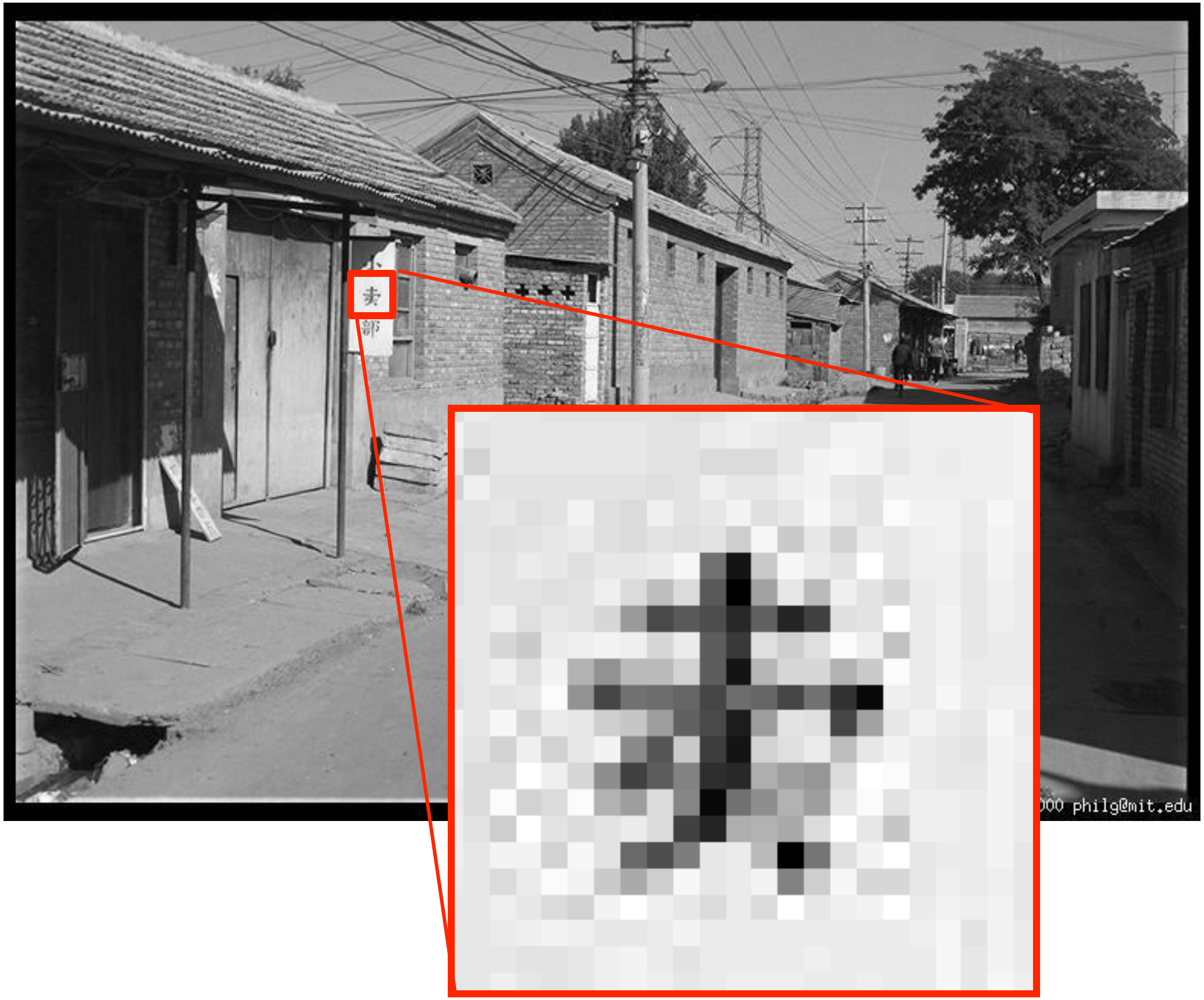
FIGURE 2.17 (a) Continuos image projected onto a sensor array. (b) Result of image sampling and quantization.



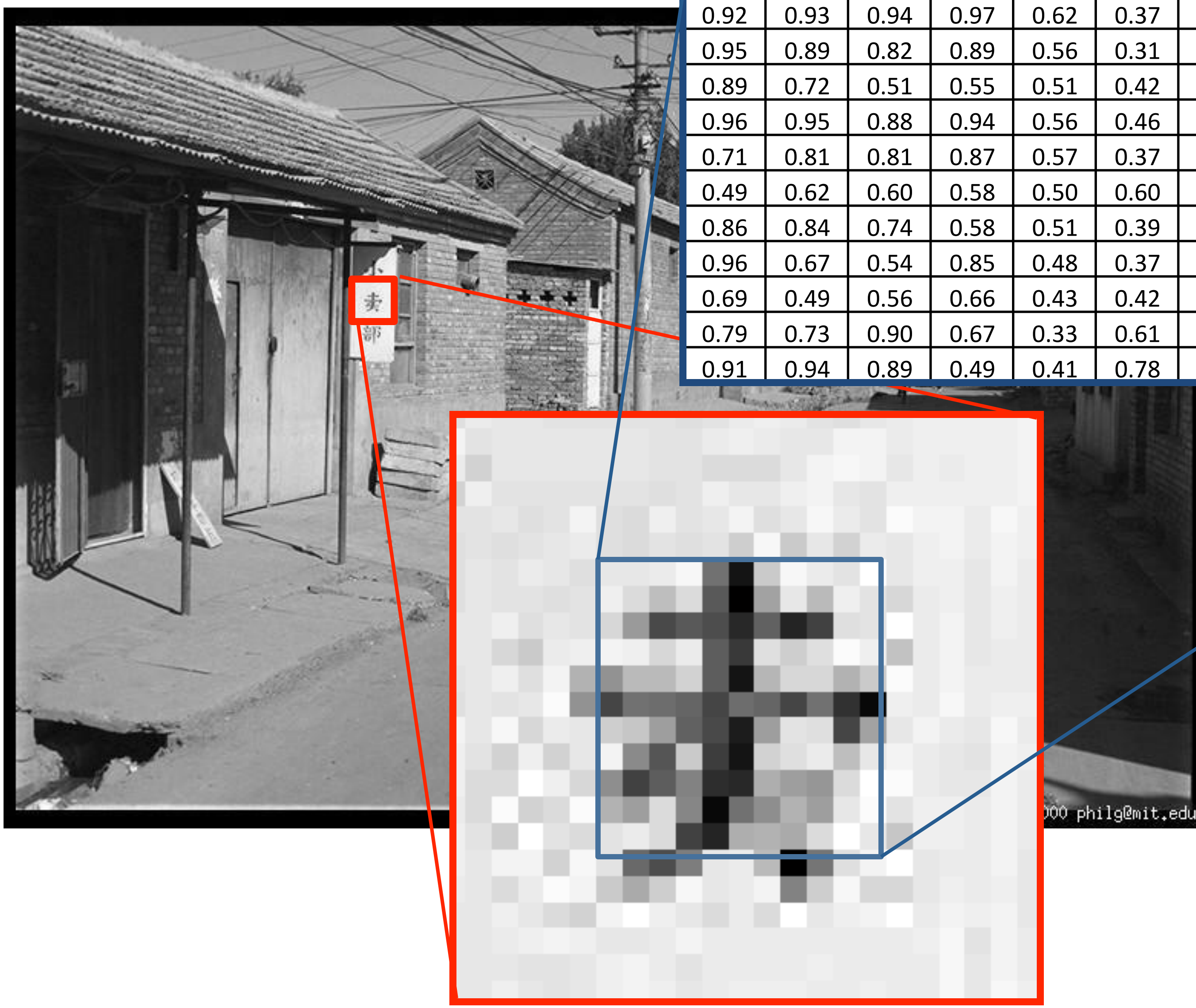
CMOS sensor

Each sensor cell records amount of light coming in at a small range of orientations

The raster image (pixel matrix)



The raster image (pixel matrix)

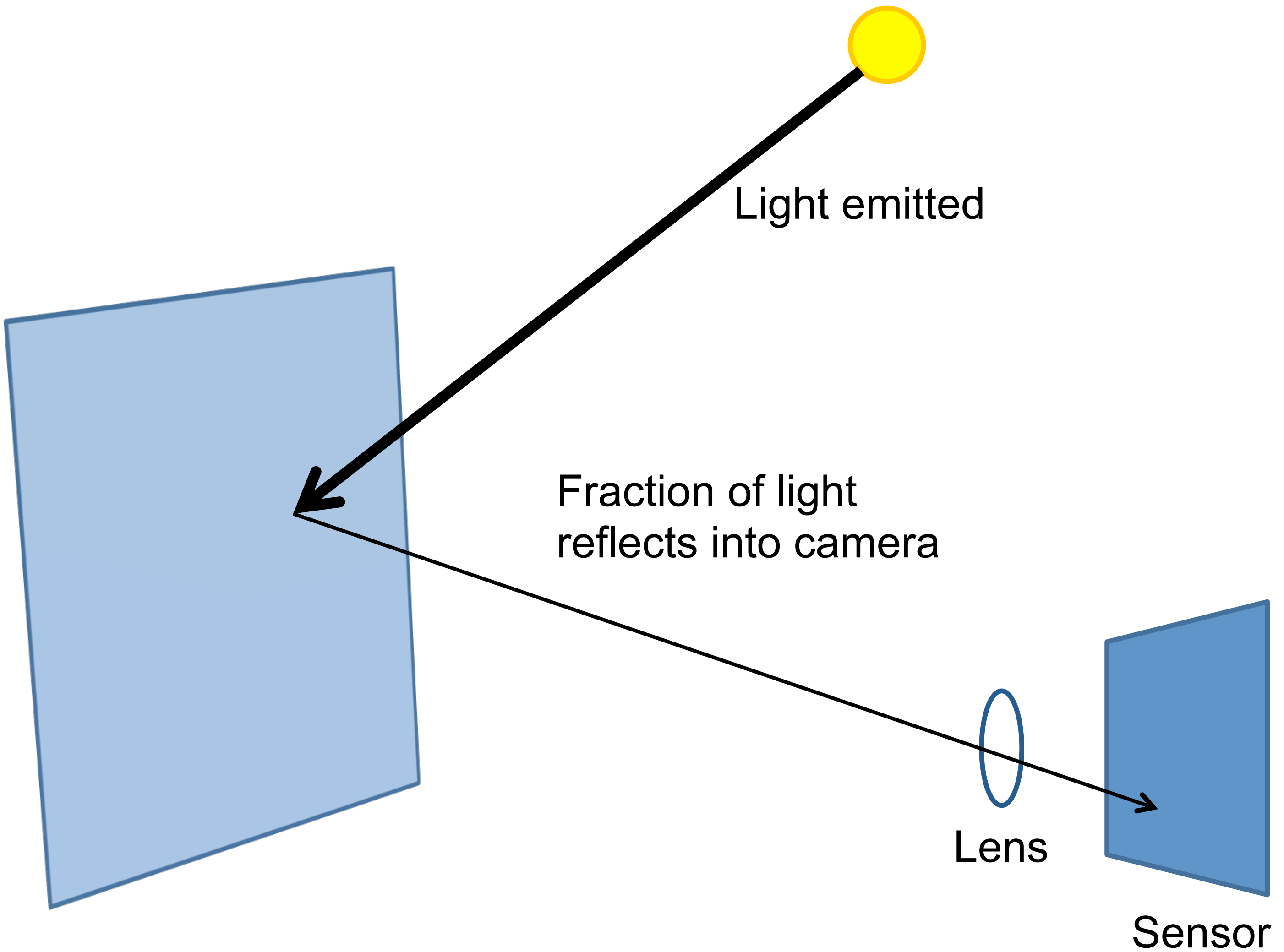


0.92	0.93	0.94	0.97	0.62	0.37	0.85	0.97	0.93	0.92	0.99
0.95	0.89	0.82	0.89	0.56	0.31	0.75	0.92	0.81	0.95	0.91
0.89	0.72	0.51	0.55	0.51	0.42	0.57	0.41	0.49	0.91	0.92
0.96	0.95	0.88	0.94	0.56	0.46	0.91	0.87	0.90	0.97	0.95
0.71	0.81	0.81	0.87	0.57	0.37	0.80	0.88	0.89	0.79	0.85
0.49	0.62	0.60	0.58	0.50	0.60	0.58	0.50	0.61	0.45	0.33
0.86	0.84	0.74	0.58	0.51	0.39	0.73	0.92	0.91	0.49	0.74
0.96	0.67	0.54	0.85	0.48	0.37	0.88	0.90	0.94	0.82	0.93
0.69	0.49	0.56	0.66	0.43	0.42	0.77	0.73	0.71	0.90	0.99
0.79	0.73	0.90	0.67	0.33	0.61	0.69	0.79	0.73	0.93	0.97
0.91	0.94	0.89	0.49	0.41	0.78	0.78	0.77	0.89	0.99	0.93

Today's class: Light and Shading

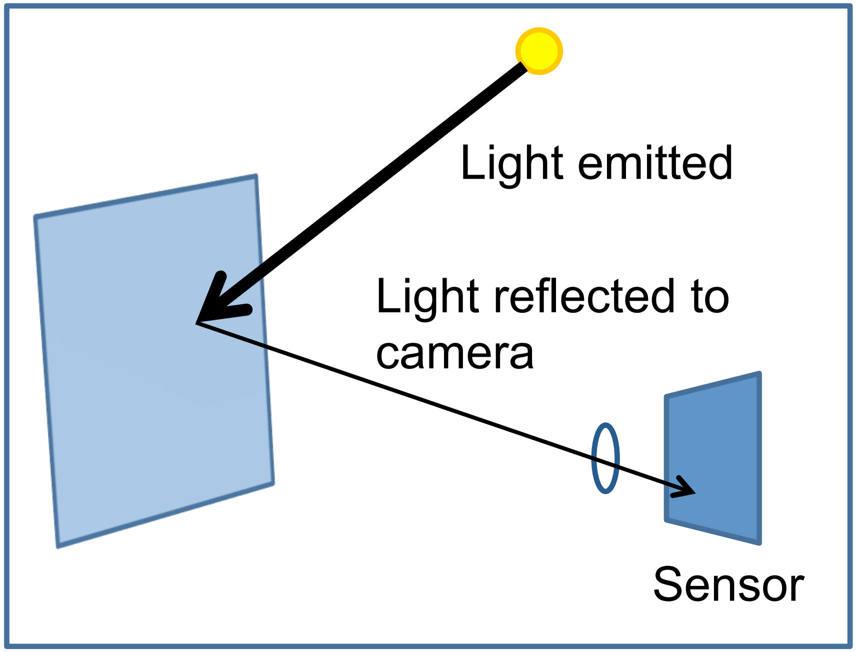
- What determines a pixel's intensity?
- What can we infer about the scene from pixel intensities?

How does a pixel get its value?



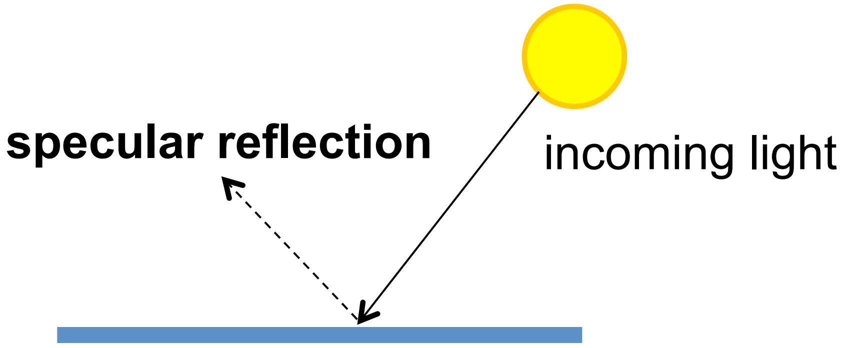
How does a pixel get its value?

- Major factors
  - Illumination strength and direction
  - Surface geometry
  - Surface material
  - Nearby surfaces
  - Camera gain/exposure

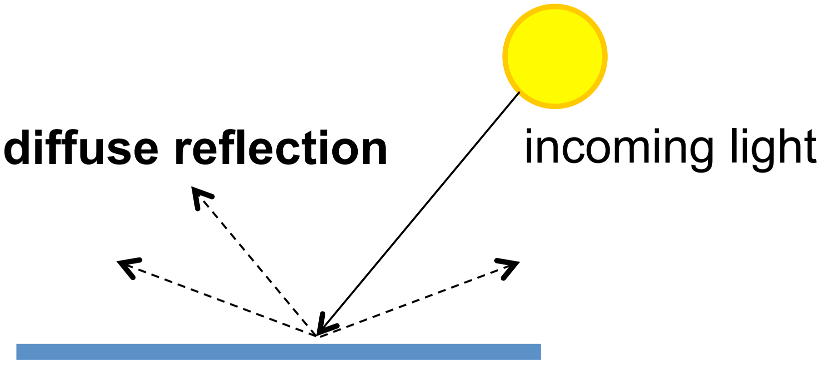


Basic models of reflection

- Specular: light bounces off at the incident angle
  - E.g., mirror

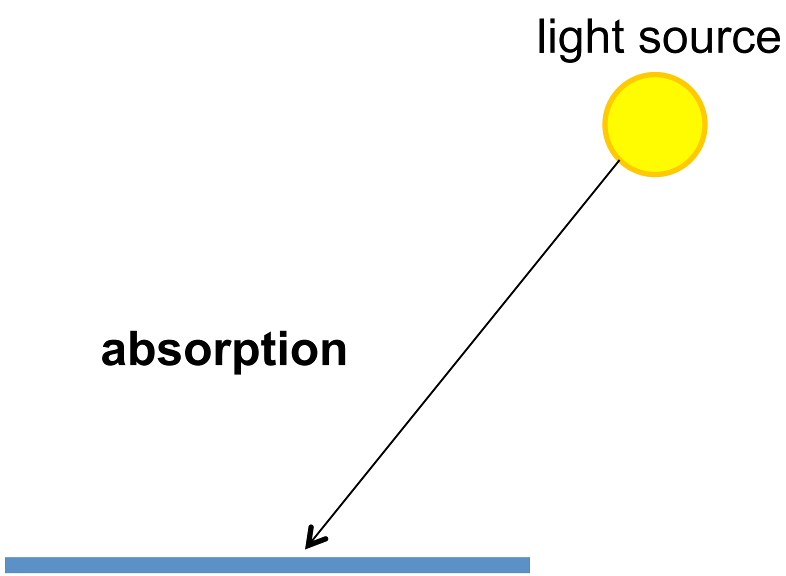


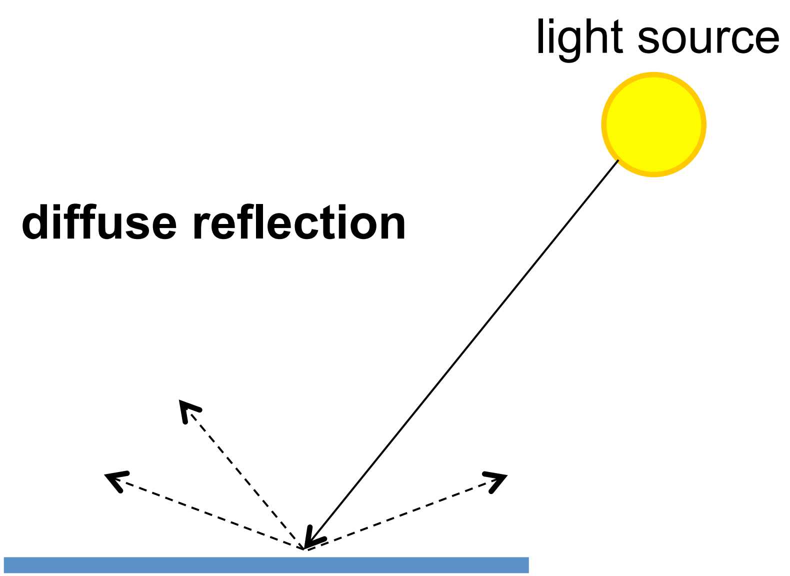
- Diffuse: light scatters in all directions
  - E.g., brick, cloth, rough wood



Diffuse reflectance model

- Some light is absorbed (function of albedo ρ)
- Remaining light is scattered (diffuse reflection)
- Examples: soft cloth, concrete, matte paints

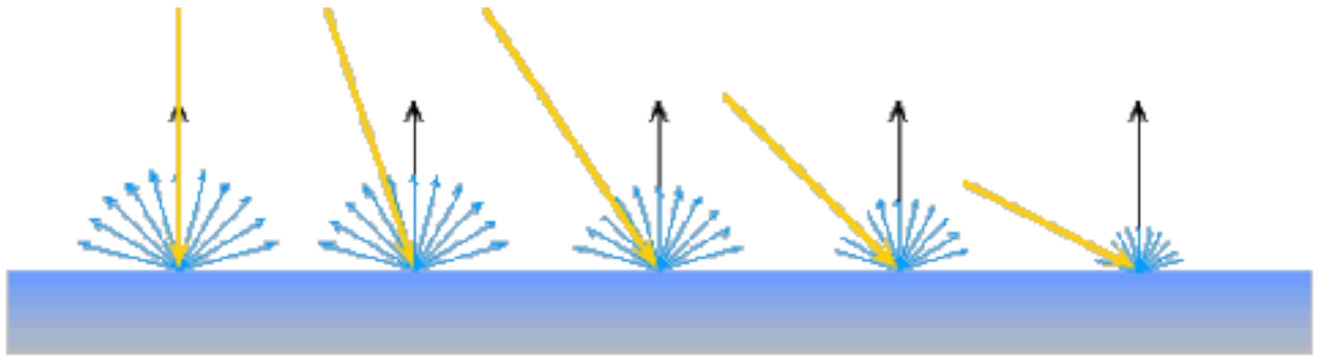




Diffuse reflection

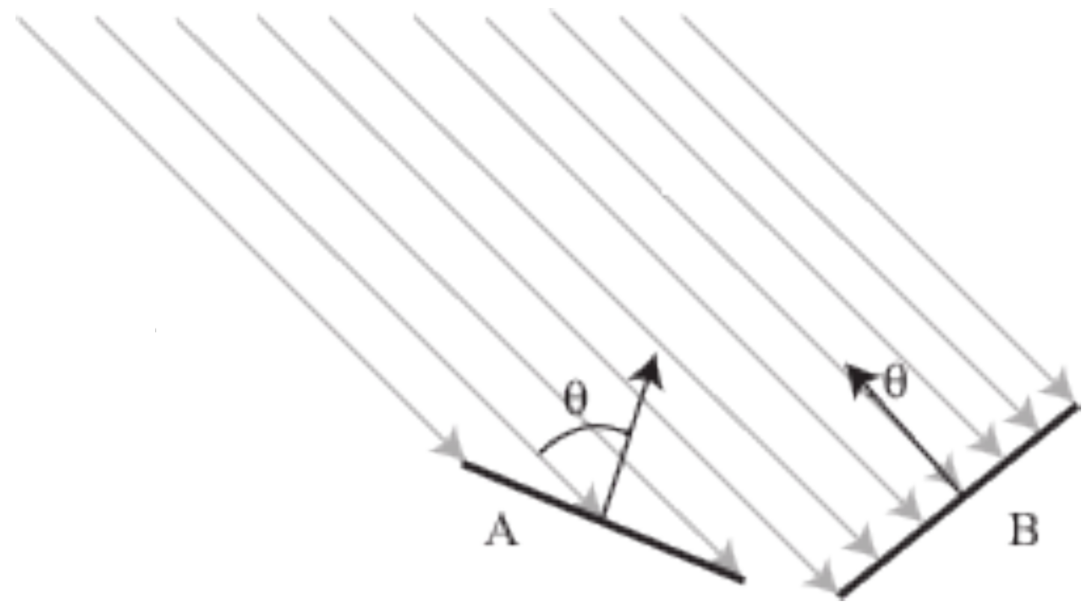
Diffuse reflection governed by **Lambert's law**

- Viewed brightness does not depend on viewing direction
- Brightness *does* depend on direction of illumination
- This is the model most often used in computer vision



Intensity and Surface Orientation

Intensity depends on illumination angle because less light comes in at oblique angles.



Slide: Forsyth

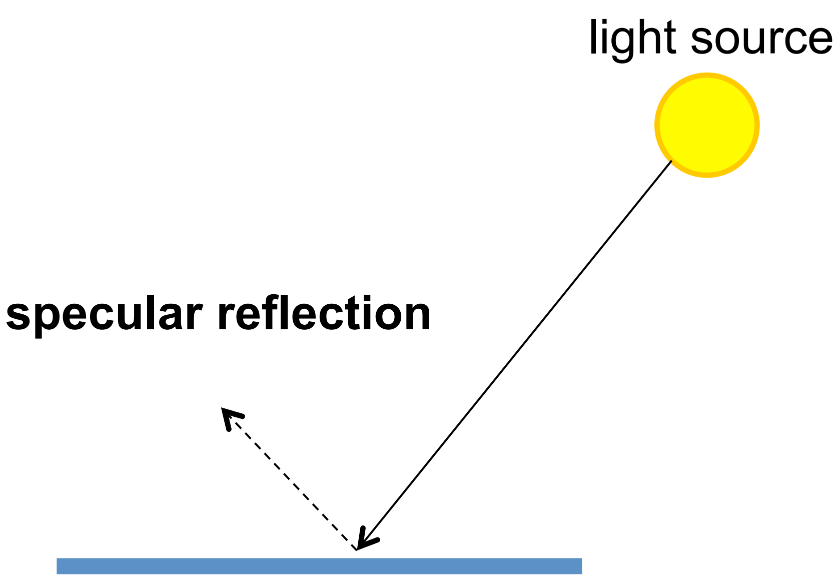

1
2

Specular Reflection

- Reflected direction depends on light orientation and surface normal
  - E.g., mirrors are fully specular
  - Most surfaces can be modeled with a mixture of diffuse and specular components

Flickr, by suzysputnik

Flickr, by piratejohnny



Most surfaces have both specular and diffuse components

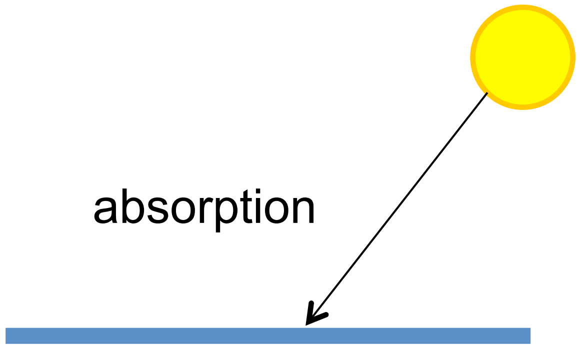
- Specularity = spot where specular reflection dominates (typically reflects light source)

Photo: northcountryhardwoodfloors.com

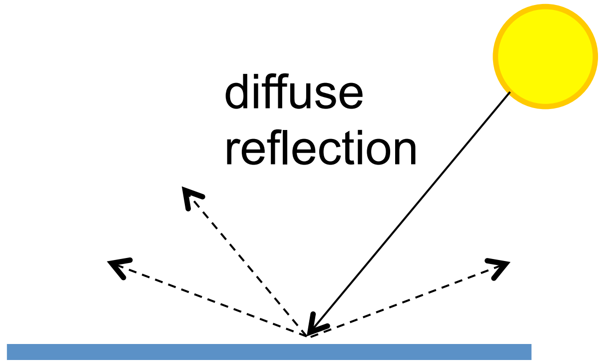
Typically, specular component is small

Recap

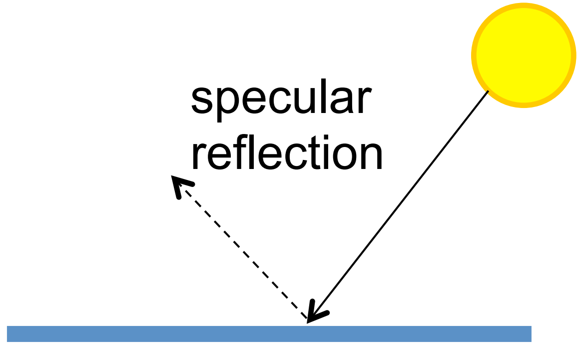
- When light hits a typical surface
  - Some light is absorbed ($1-\rho$)
    - More absorbed for low albedos



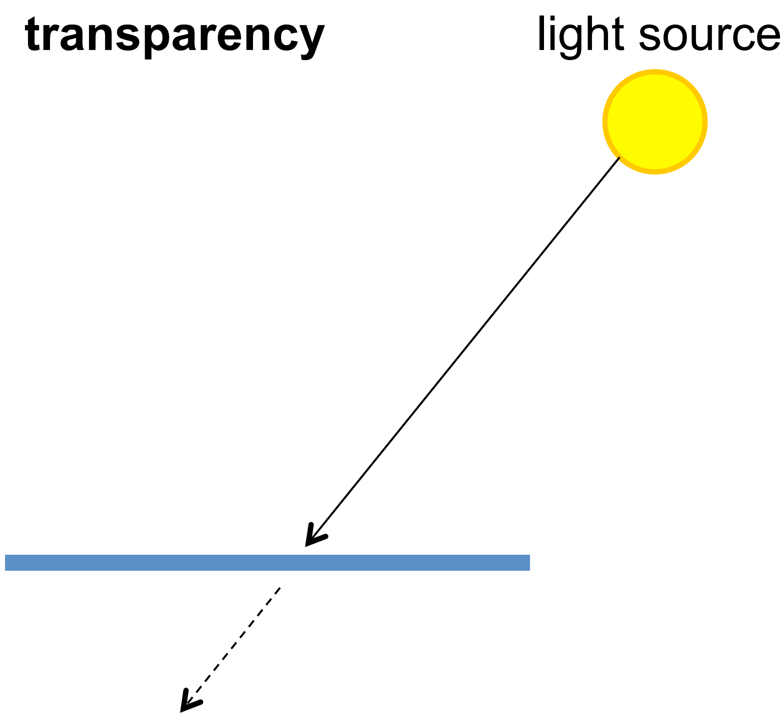
  - Some light is reflected diffusely
    - Independent of viewing direction

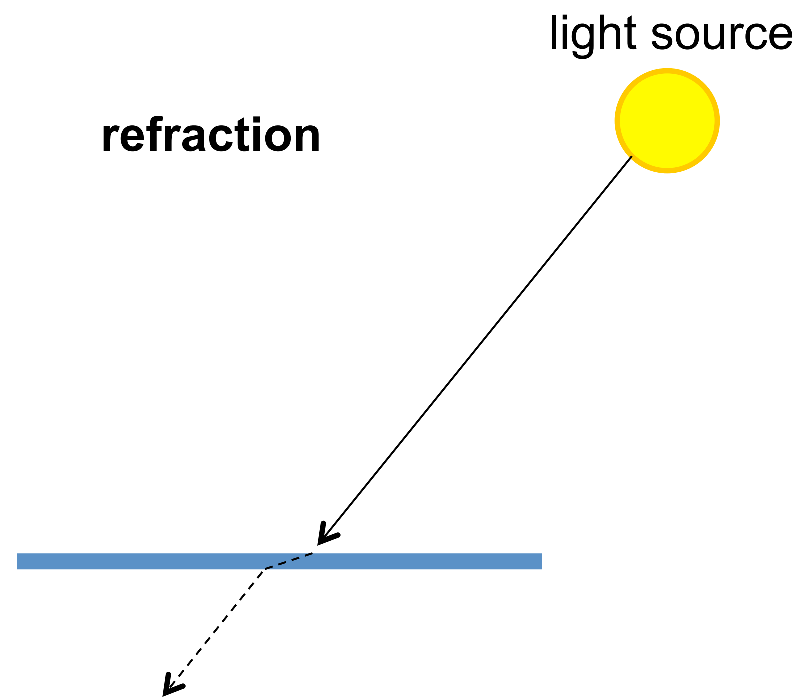


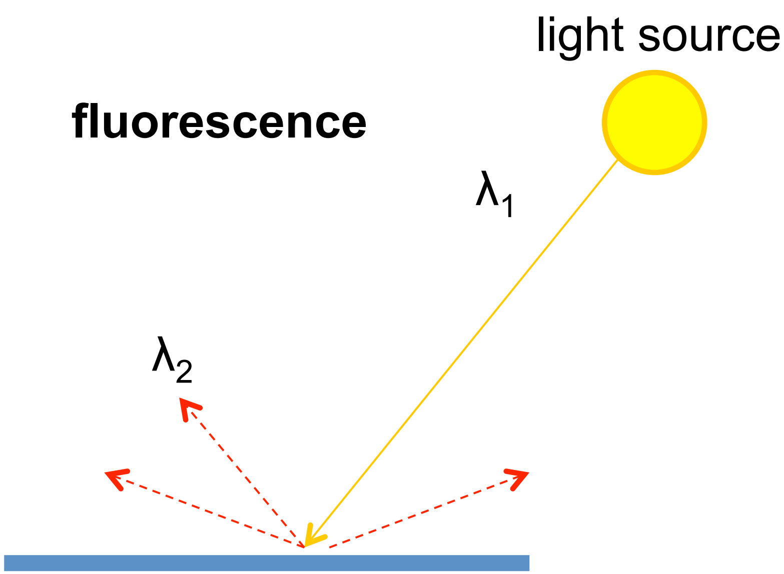
  - Some light is reflected specularly
    - Light bounces off (like a mirror), depends on viewing direction

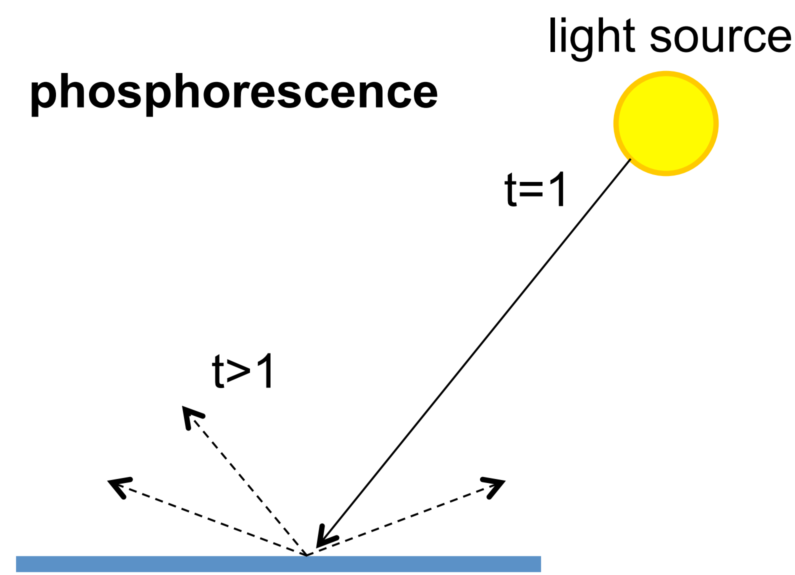


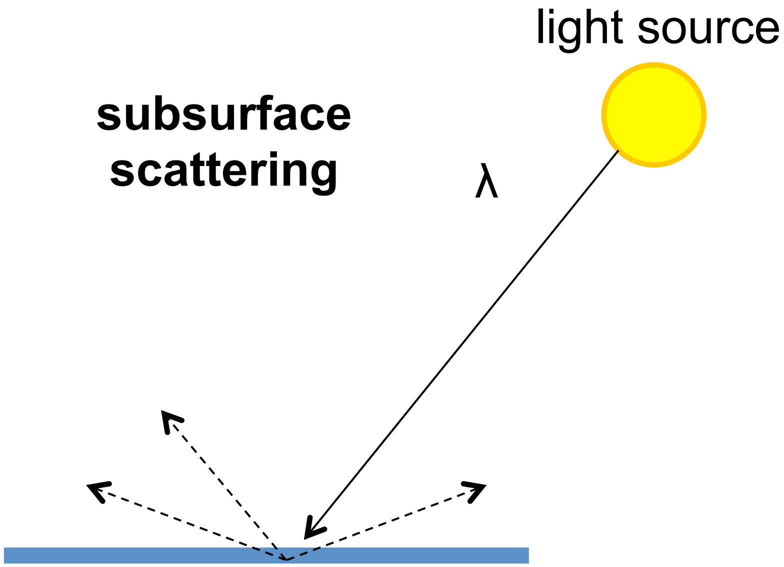
Other possible effects





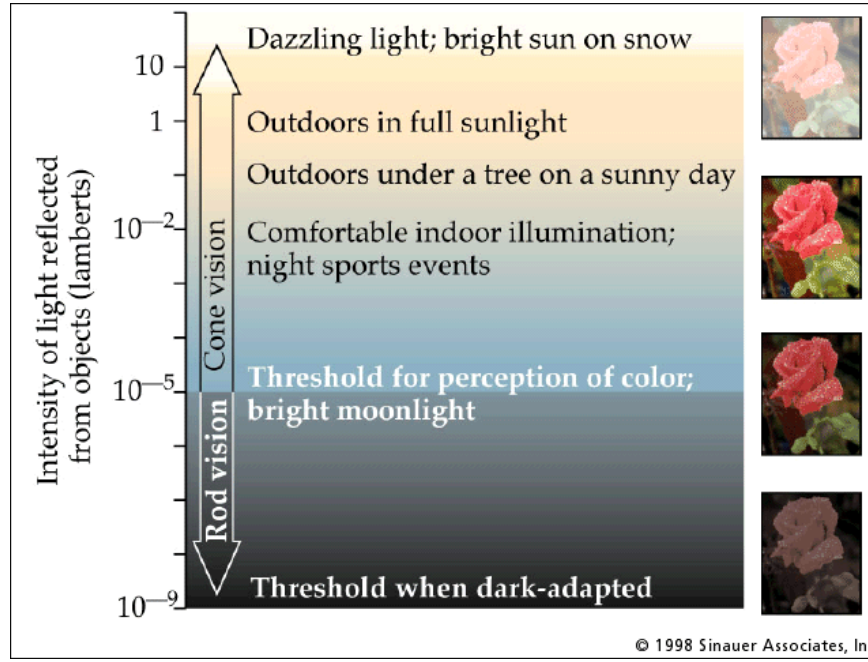

light source
fluorescence
λ_1
λ_2

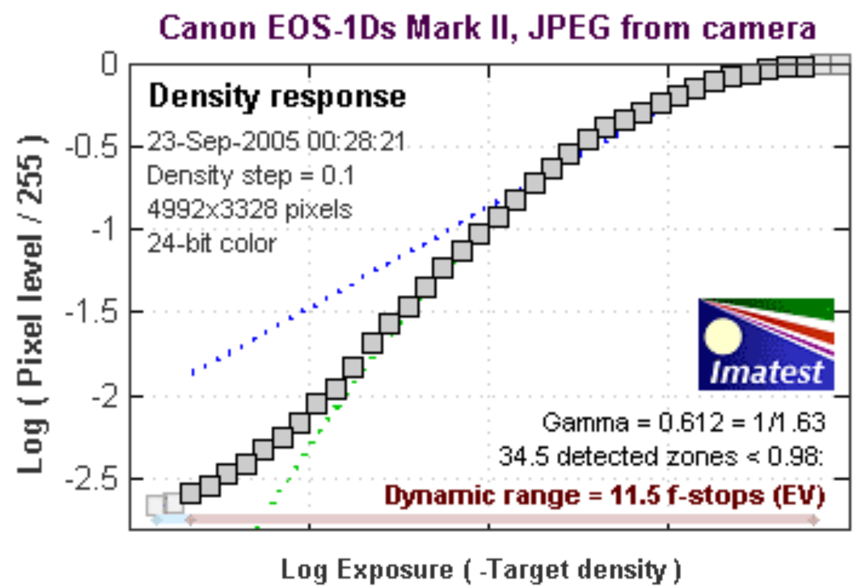

light source
phosphorescence
t=1
t>1


light source
subsurface
scattering
λ

Dynamic range and camera response

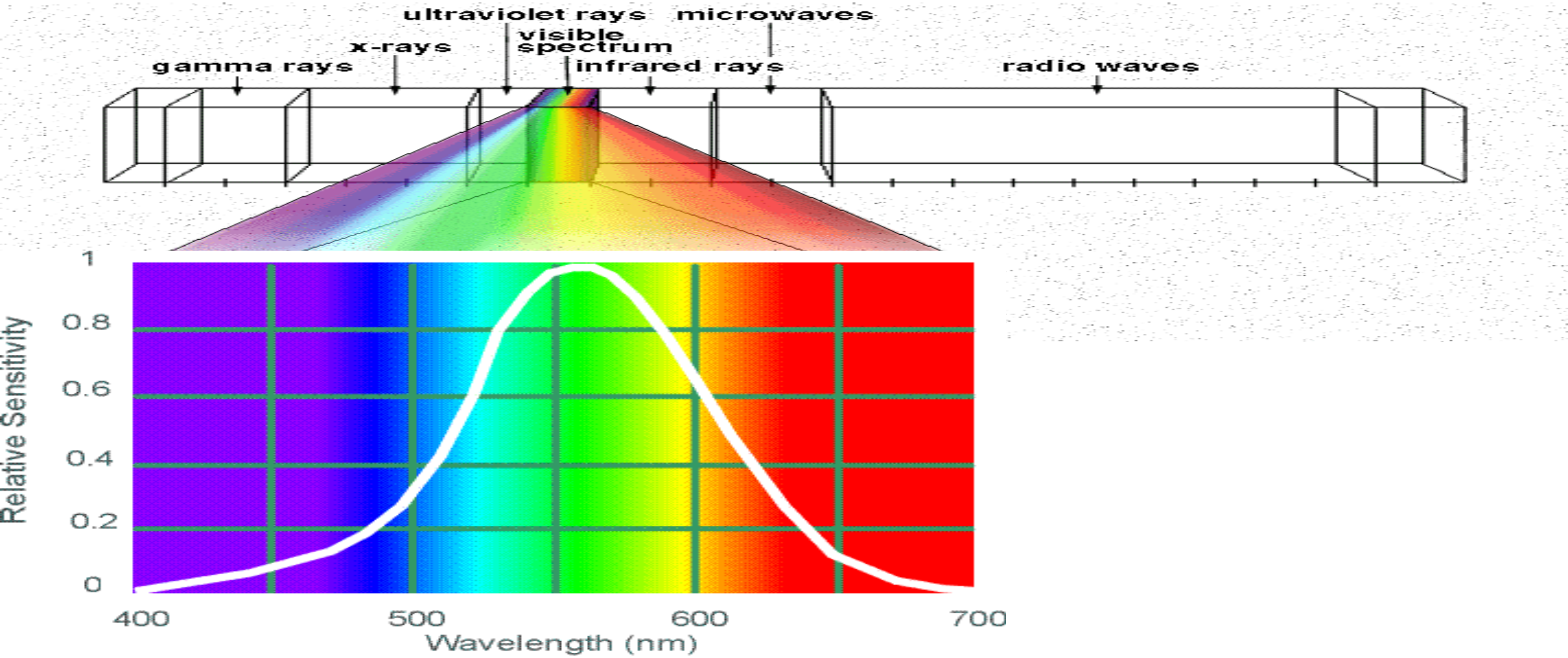
- Typical scenes have a huge dynamic range
- Camera response is roughly linear in the mid range (15 to 240) but non-linear at the extremes
  - called saturation or undersaturation





Color

Light is composed of a spectrum of wavelengths

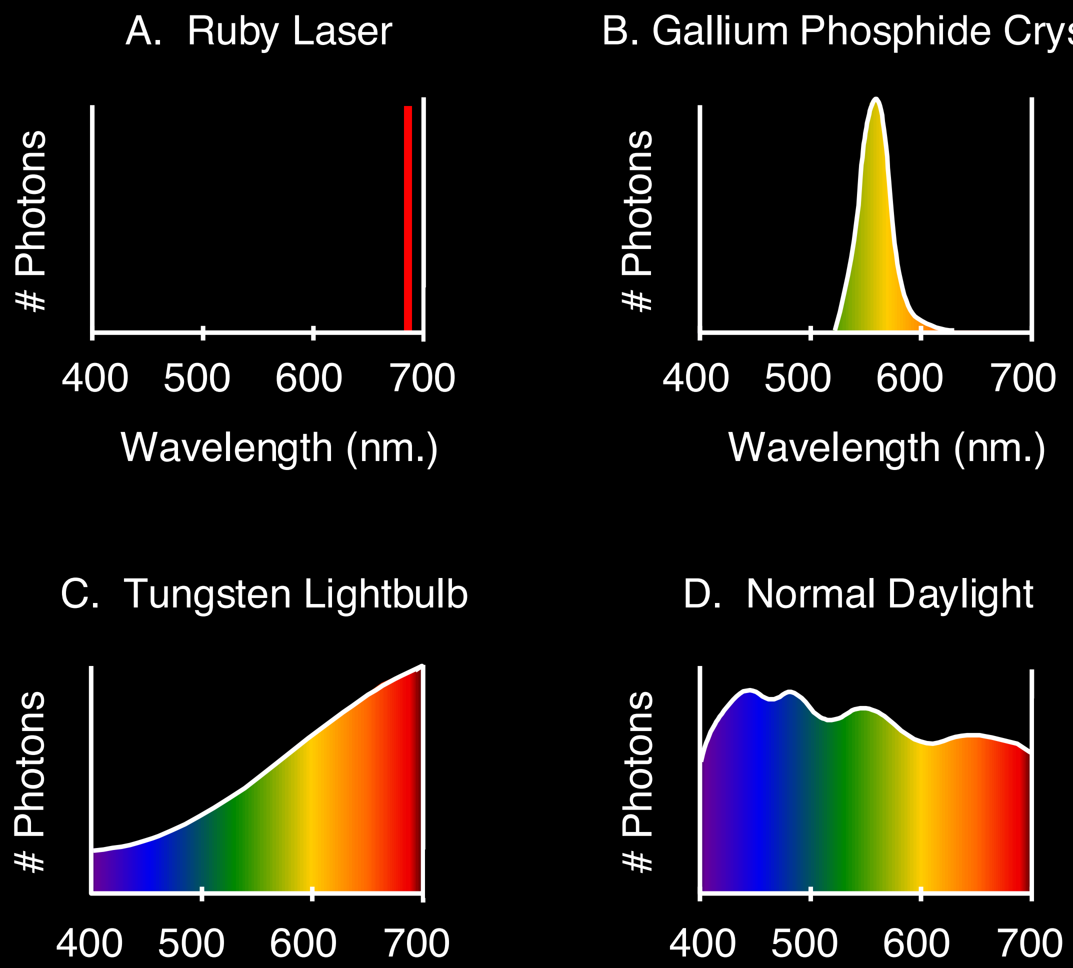


Human Luminance Sensitivity Function

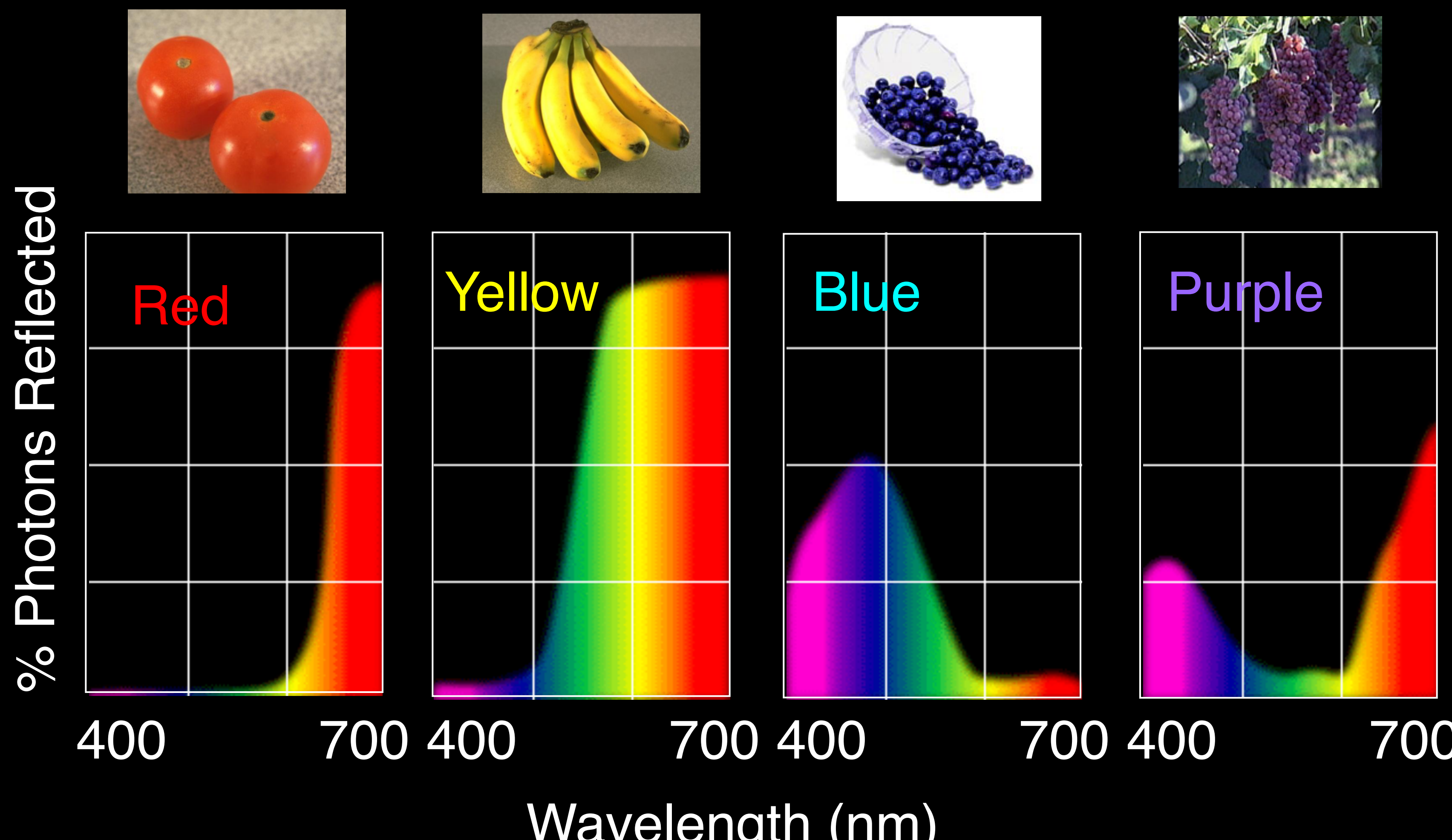
Slide Credit: Efros

http://www.yorku.ca/eye/photopik.htm

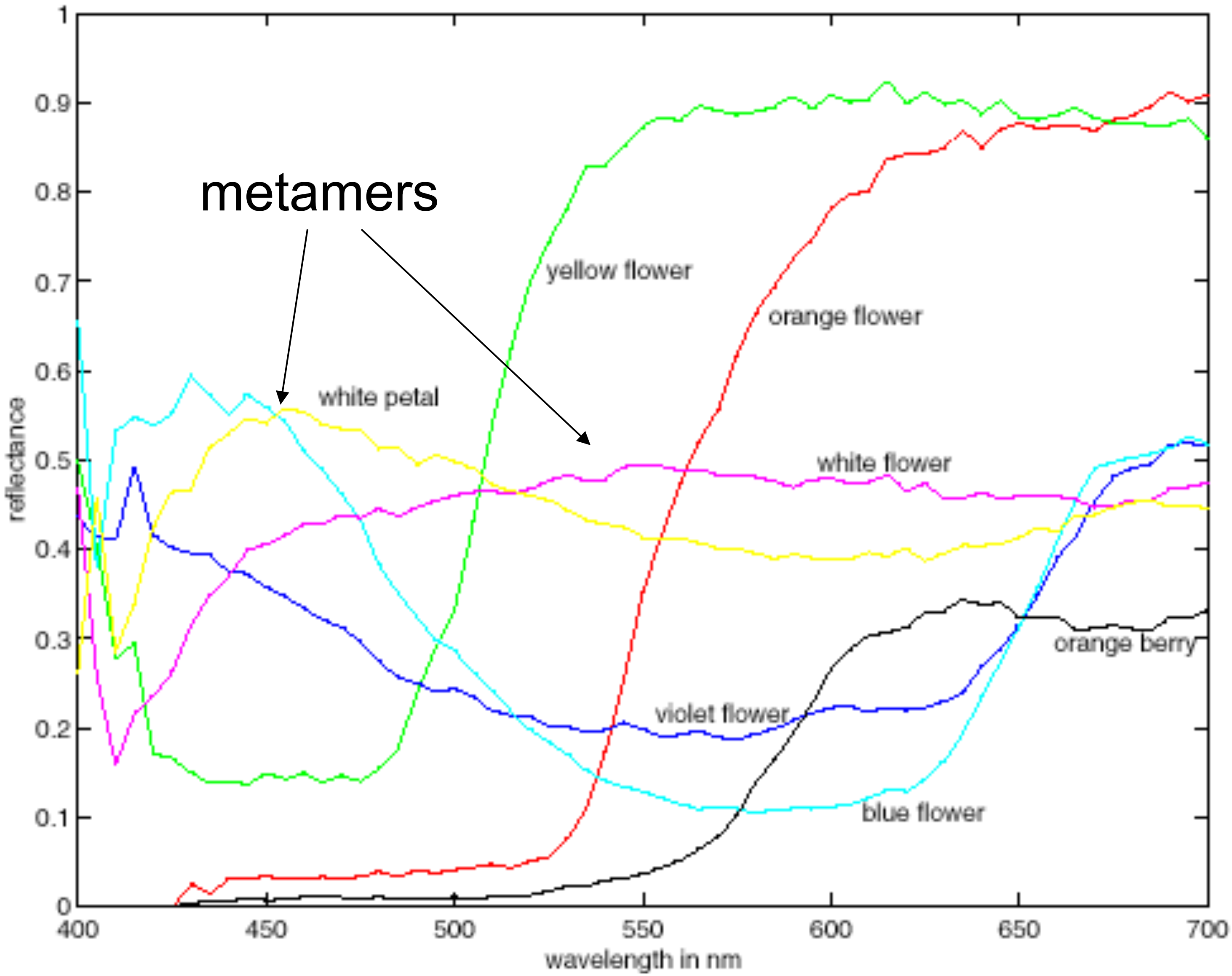
Some examples of the spectra of light sources



Some examples of the reflectance spectra of surfaces

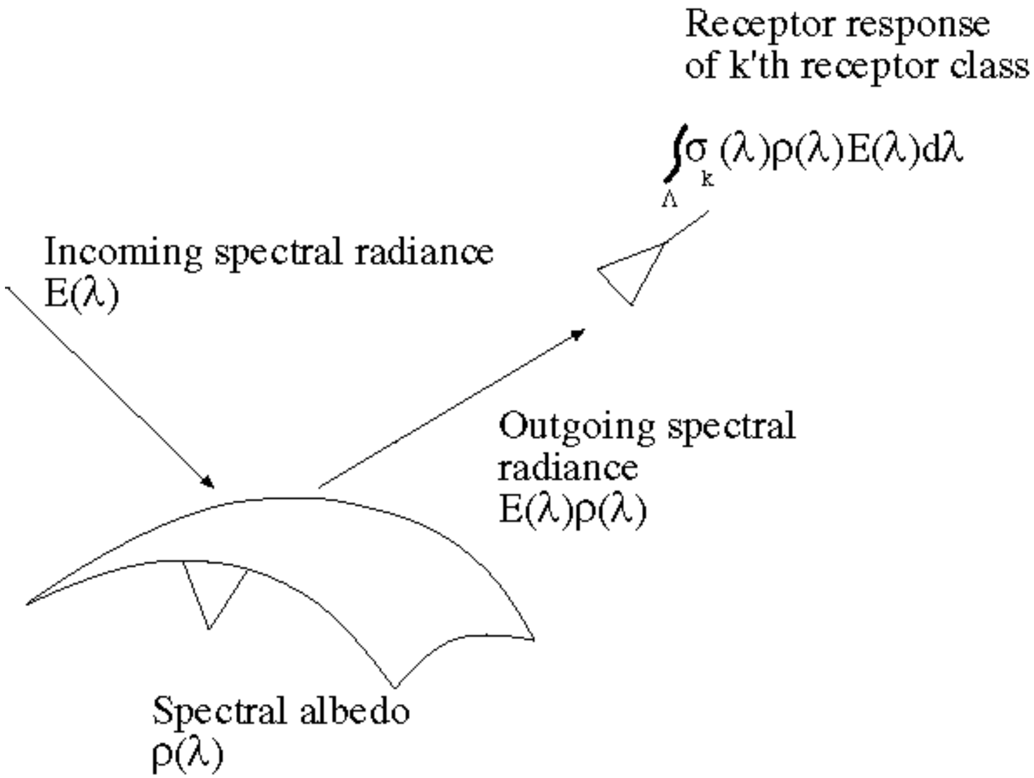


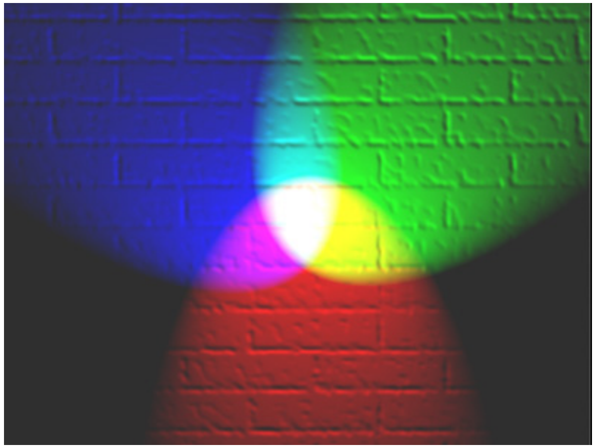
More spectra



The color of objects

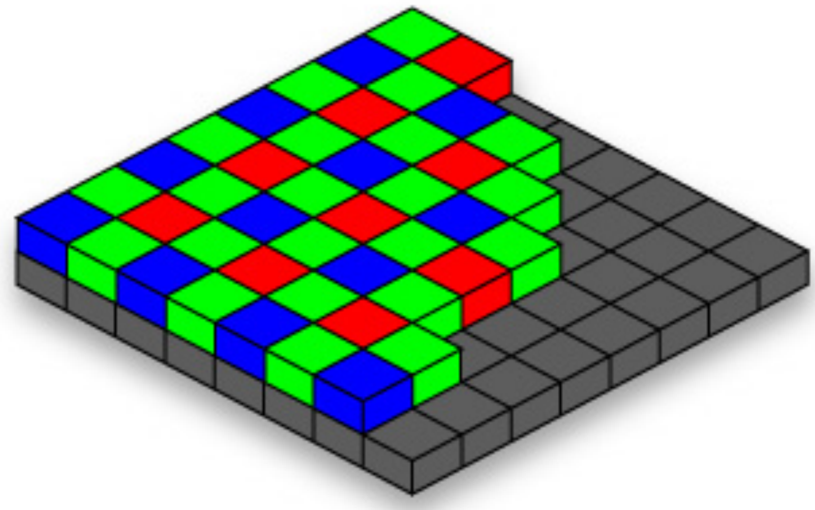
- Colored light arriving at the camera involves two effects
  - The color of the light source (illumination + inter-reflections)
  - The color of the surface



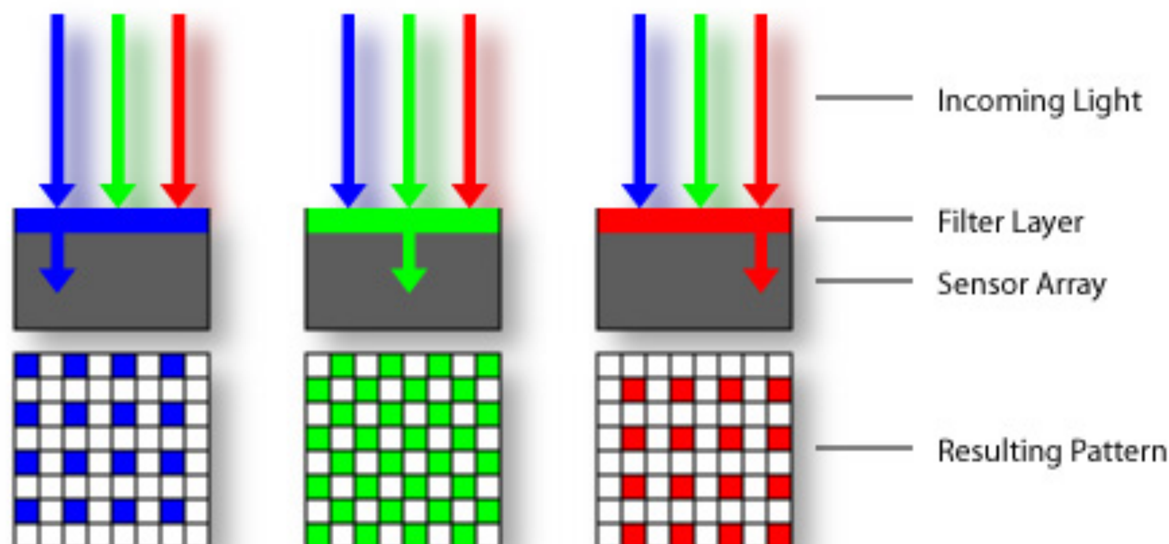


Slide: Forsyth

Color Sensing: Bayer Grid



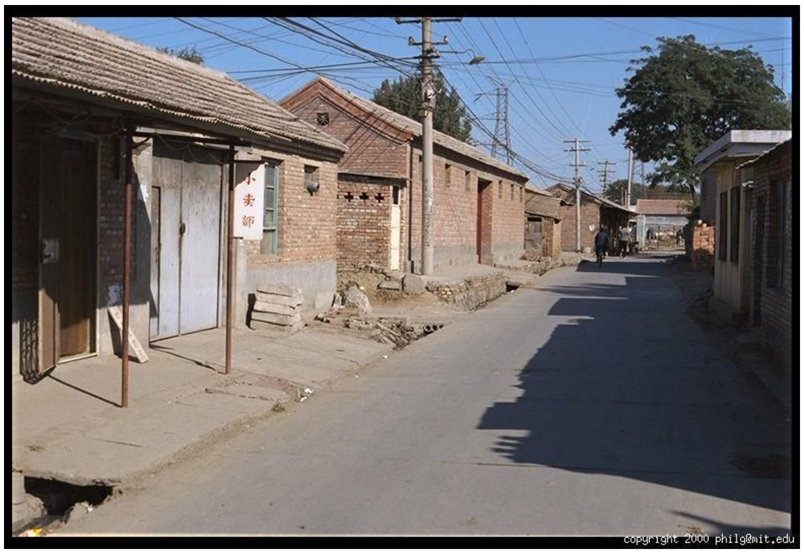
- Estimate RGB at each cell from neighboring values



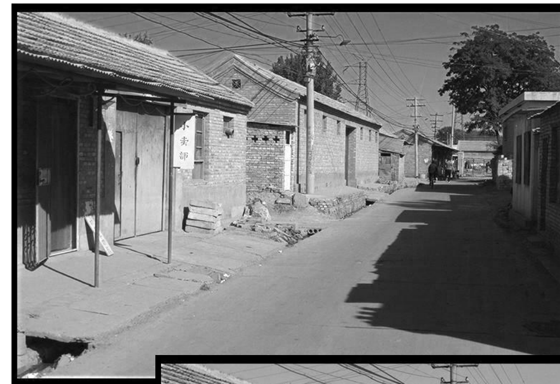
http://en.wikipedia.org/wiki/Bayer_filter

Slide by Steve Seitz

Color Image

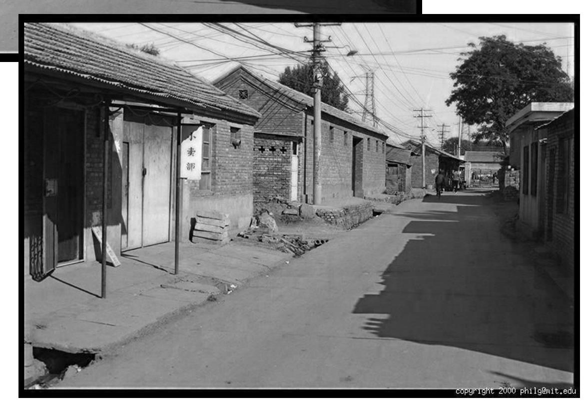


R



G

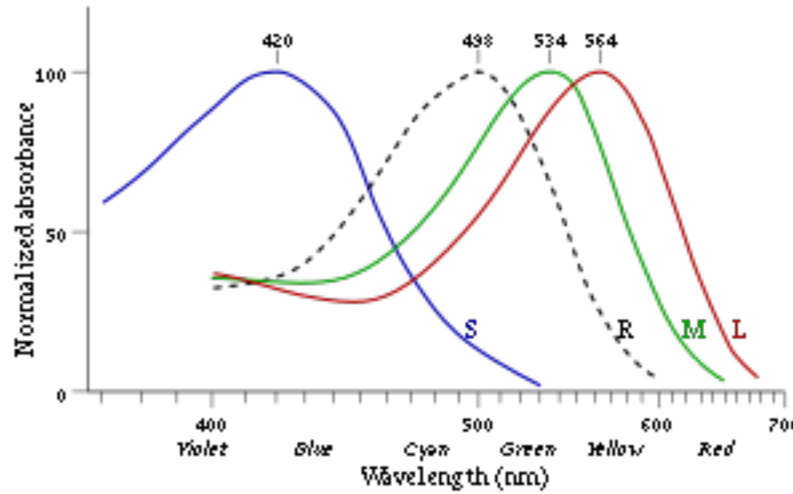
B



Why RGB?

If light is a spectrum, why are images RGB?

Human color receptors

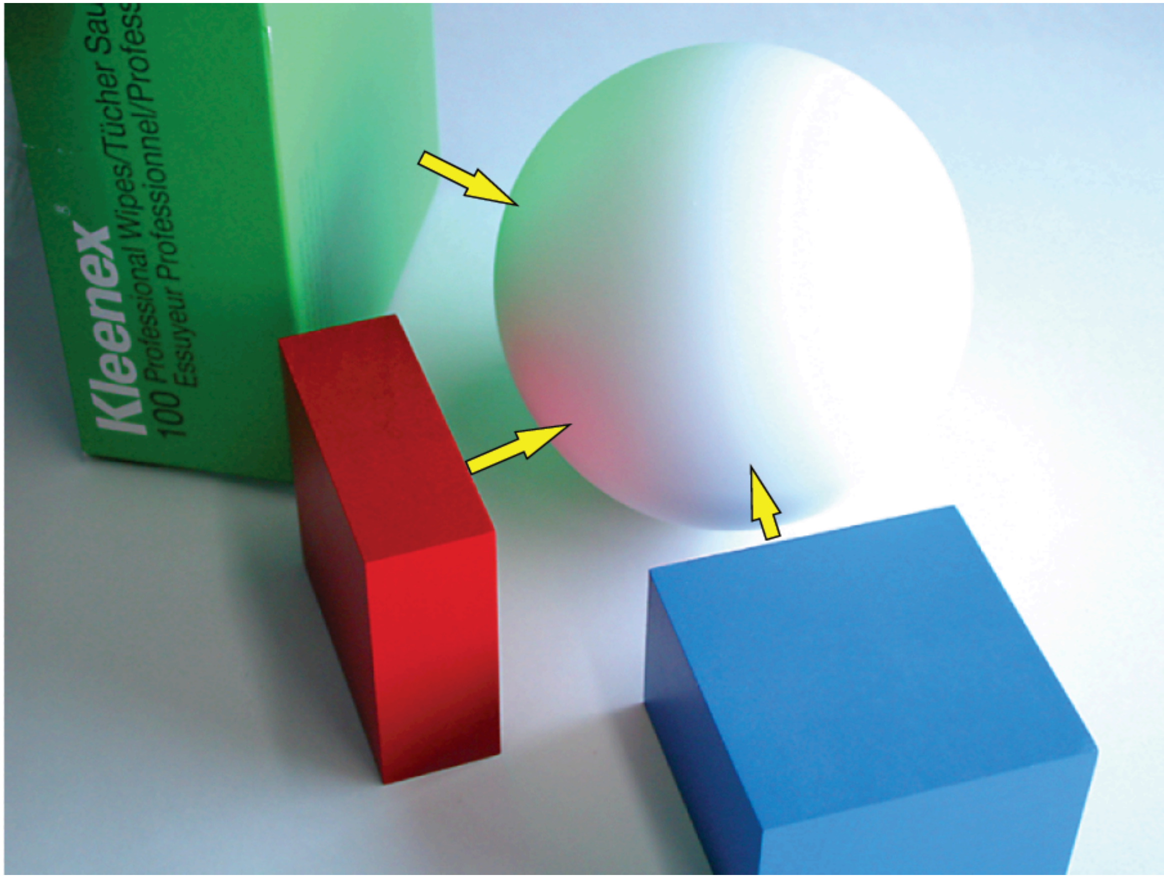


- Long (red), Medium (green), and Short (blue) cones, plus intensity rods

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

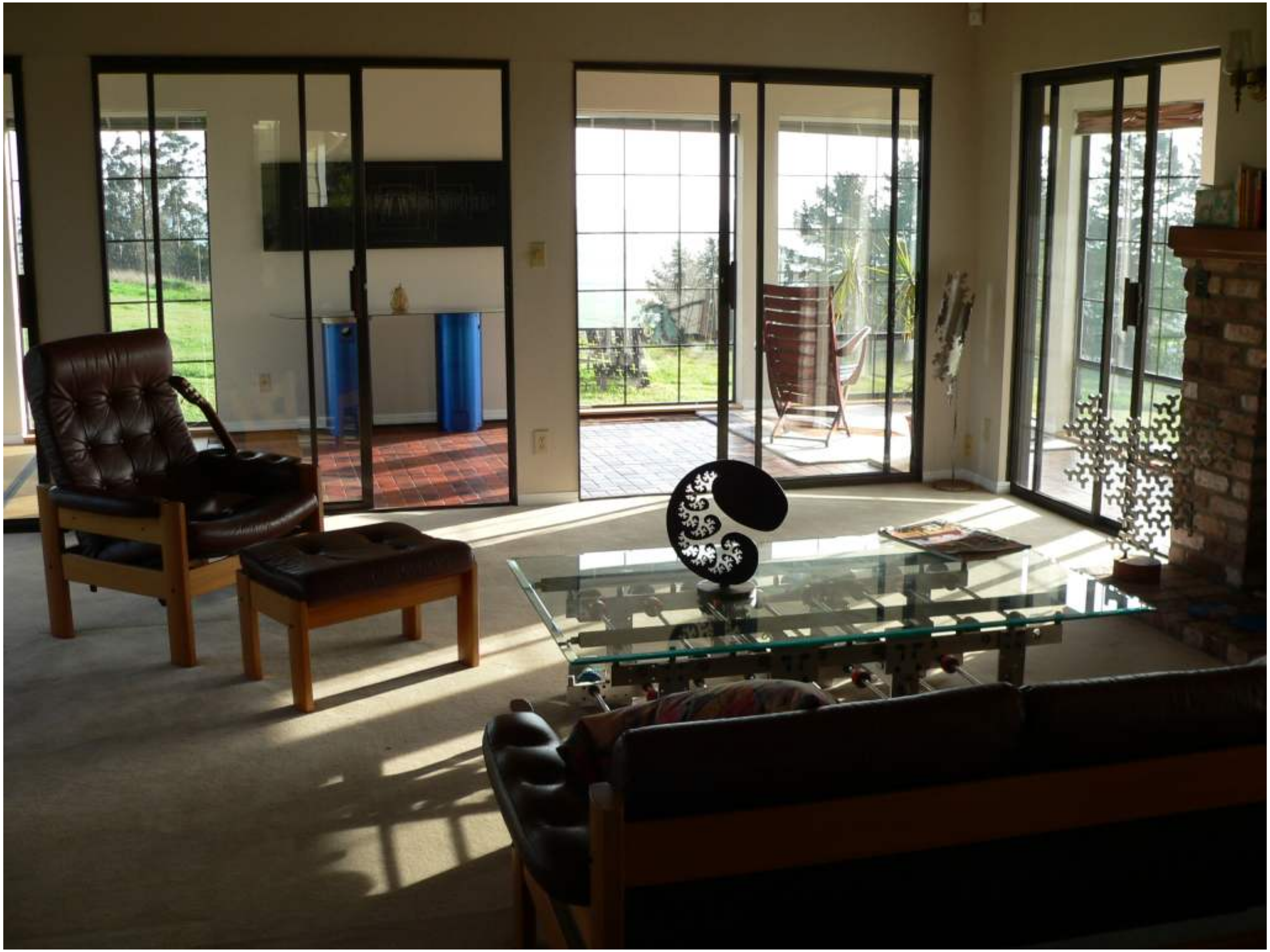
So far: light→surface→camera

- Called a local illumination model
- But much light comes from surrounding surfaces

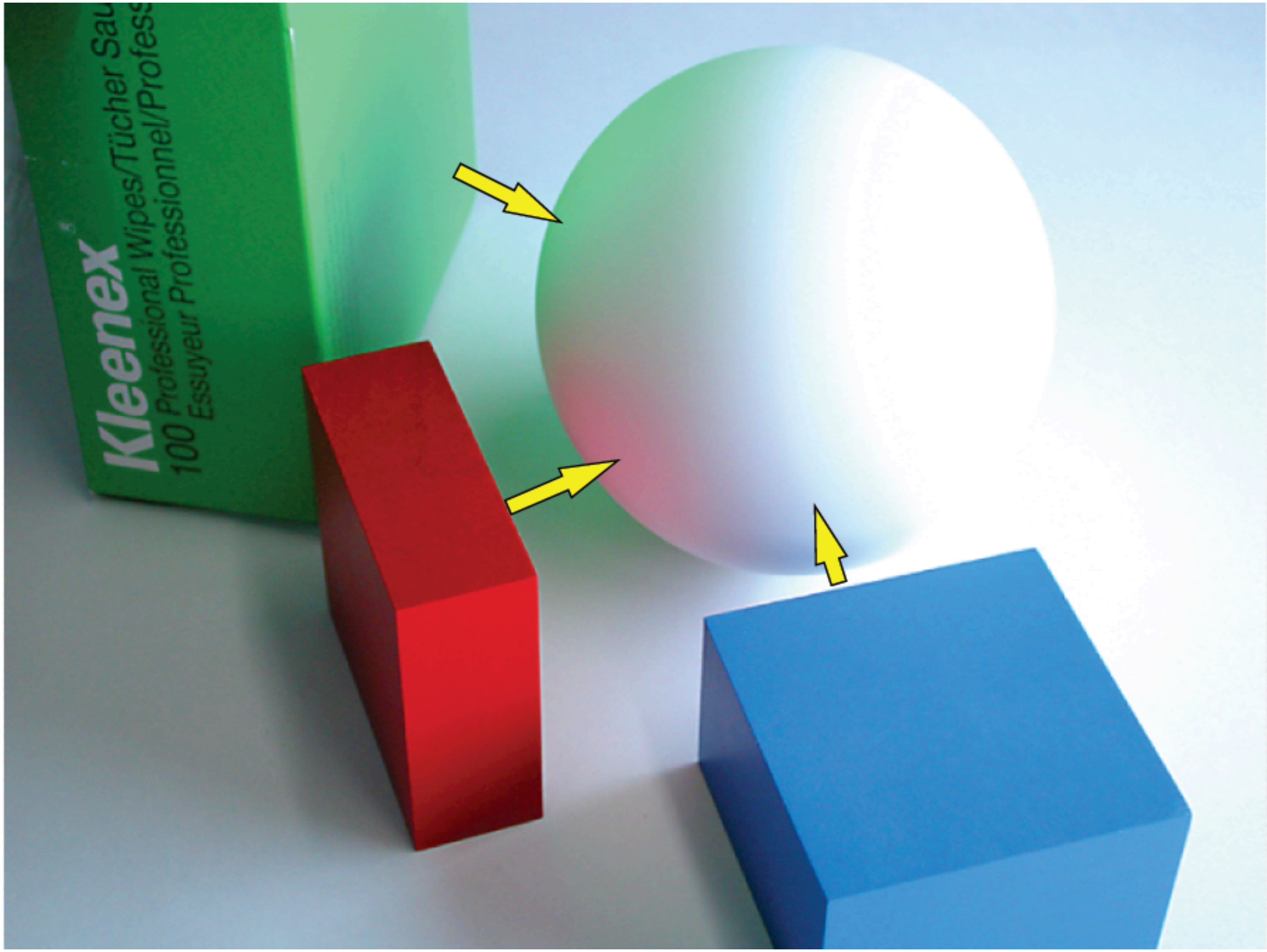


From Koenderink slides on image texture and the flow of light

Inter-reflection is a major source of light



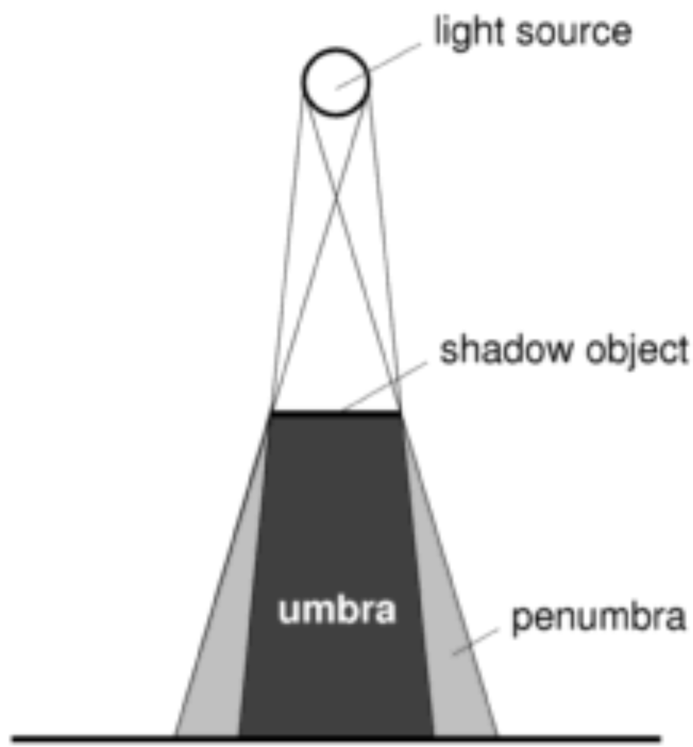
Inter-reflection affects the apparent color of objects



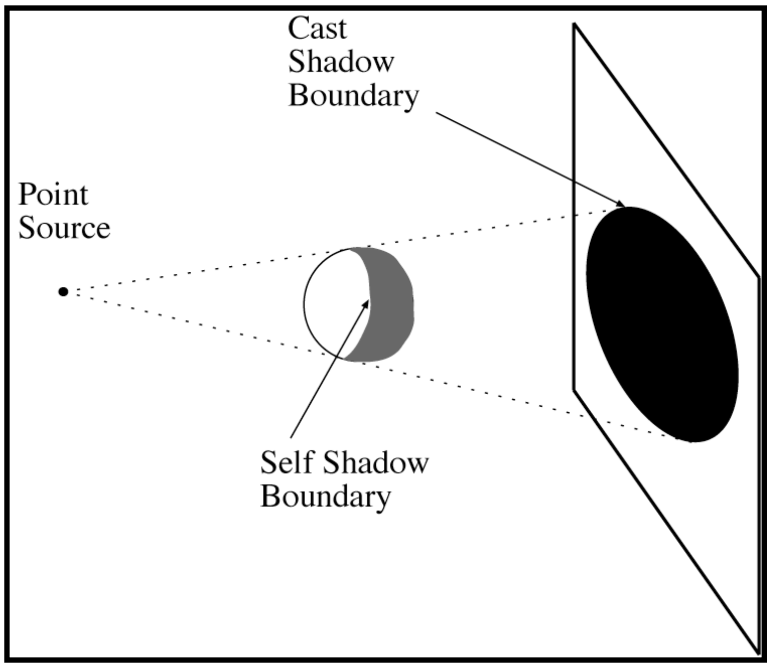
From Koenderink slides on image texture and the flow of light

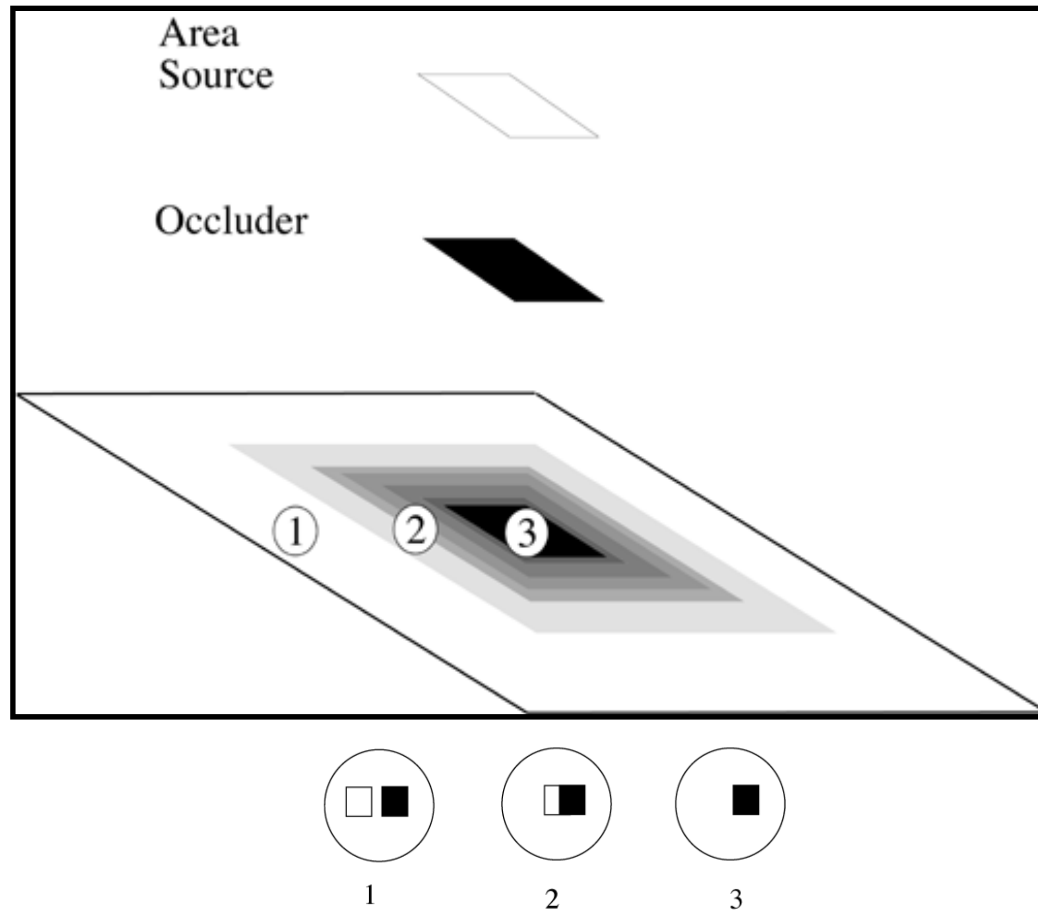
Scene surfaces also cause shadows

- Shadow: reduction in intensity due to a blocked source



Shadows


Cast
Shadow
Boundary
Point
Source
Self Shadow
Boundary


Area
Source
Occluder
1
2
3
1
2
3

Models of light sources

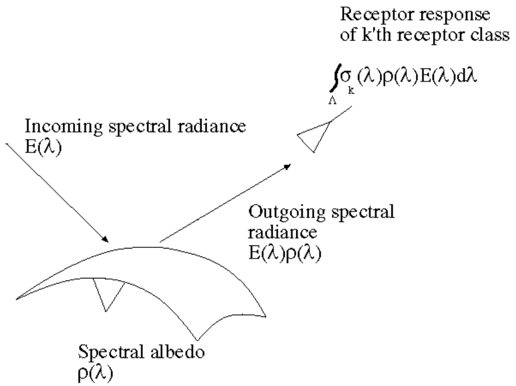
- Distant point source
  - One illumination direction
  - E.g., sun
- Area source
  - E.g., white walls, diffuser lamps, sky
- Ambient light
  - Substitute for dealing with interreflections
- Global illumination model
  - Account for interreflections in modeled scene

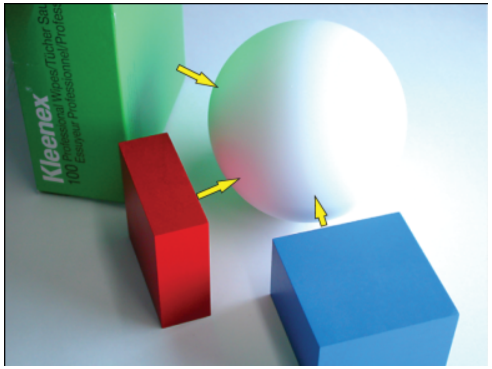
Recap

Possible factors: albedo, shadows, texture, specularities, curvature, lighting direction

Things to remember

- Important terms: diffuse/specular reflectance, albedo, umbra/penumbra
- Observed intensity depends on light sources, geometry/material of reflecting surface, surrounding objects, camera settings
- Objects cast light and shadows on each other





Thank you

- Next class: Image Processing