Computer Vision – Lecture 1

Prof. Rob Fergus

What is Computer Vision?

- *Vision* is about discovering from images what is present in the scene and where it is.

- In *Computer Vision* a camera (or several cameras) is linked to a computer. The computer interprets images of a real scene to obtain information useful for tasks such as navigation, manipulation and recognition.

The goal of computer vision

ap b

What we see

0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

What a computer sees

What is Computer Vision NOT?

- **Image processing**: image enhancement, image restoration, image compression. Take an image and process it to produce a new image which is, in some way, more desirable.

- **Computational Photography**: extending the capabilities of digital cameras through the use of computation to enable the capture of enhanced or entirely novel images of the world. (See my other course)

Why study it?

- Replicate human vision to allow a machine to see:
  - Central to that problem of Artificial Intelligence
  - Many industrial applications

- Gain insight into how we see
  - Vision is explored extensively by neuroscientists to gain an understanding of how the brain operates (e.g. the Center for Neural Science at NYU)

Applications

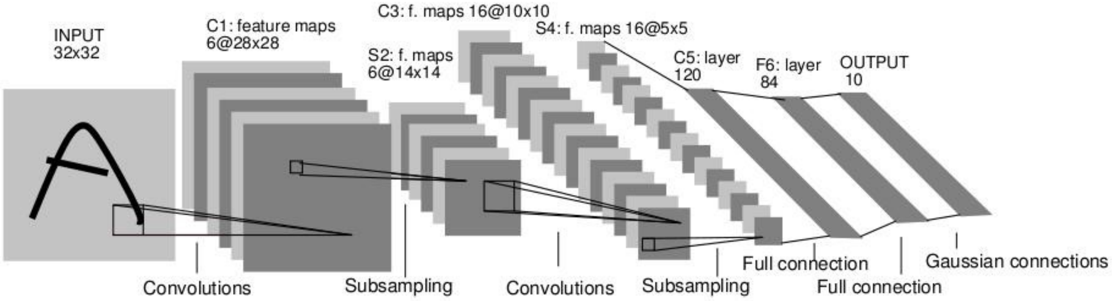
- Until ~6-7 years ago, mainly niche applications
- Now huge number of uses
  - Huge number of startups & companies, e.g. 240 @ CVPR2017 conference
- Key perceptual input for Artificial Intelligence
- Industrial robotics / inspection

  e.g. light bulbs, electronic circuits
- Self driving cars
- Security

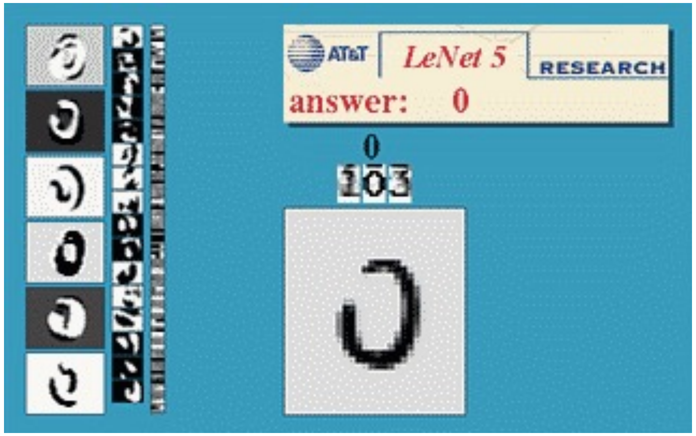
  e.g. facial recognition in airports
- Mission critical for Internet Companies
  - Google, Facebook, etc.

Convolutional Neural Network

- Developed by Yann LeCun (NYU faculty)
- Neural network with specialized connectivity structure.



- Early 1990's: Handwritten Digit recognition, License plate recognition.

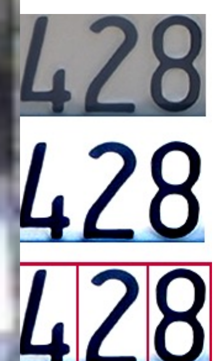


At the time, 1/3 of all checks written in US were read by this system

Convolutional Neural Network

- Developed by Yann LeCun (NYU faculty)
- Neural network with specialized connectivity structure.

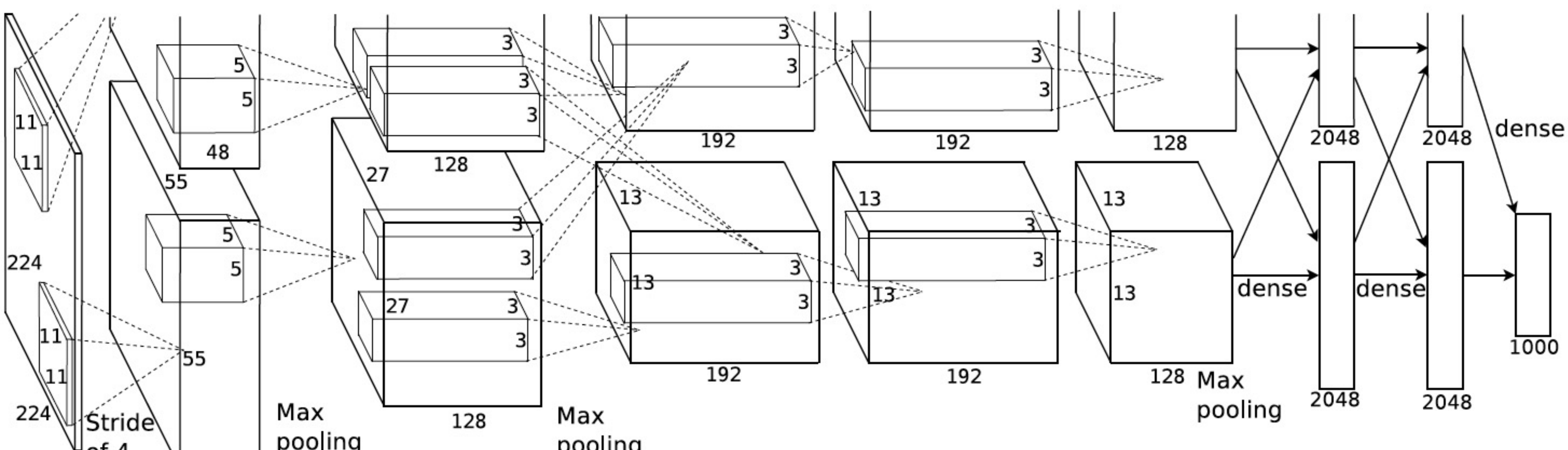
- late recognition.



by this system

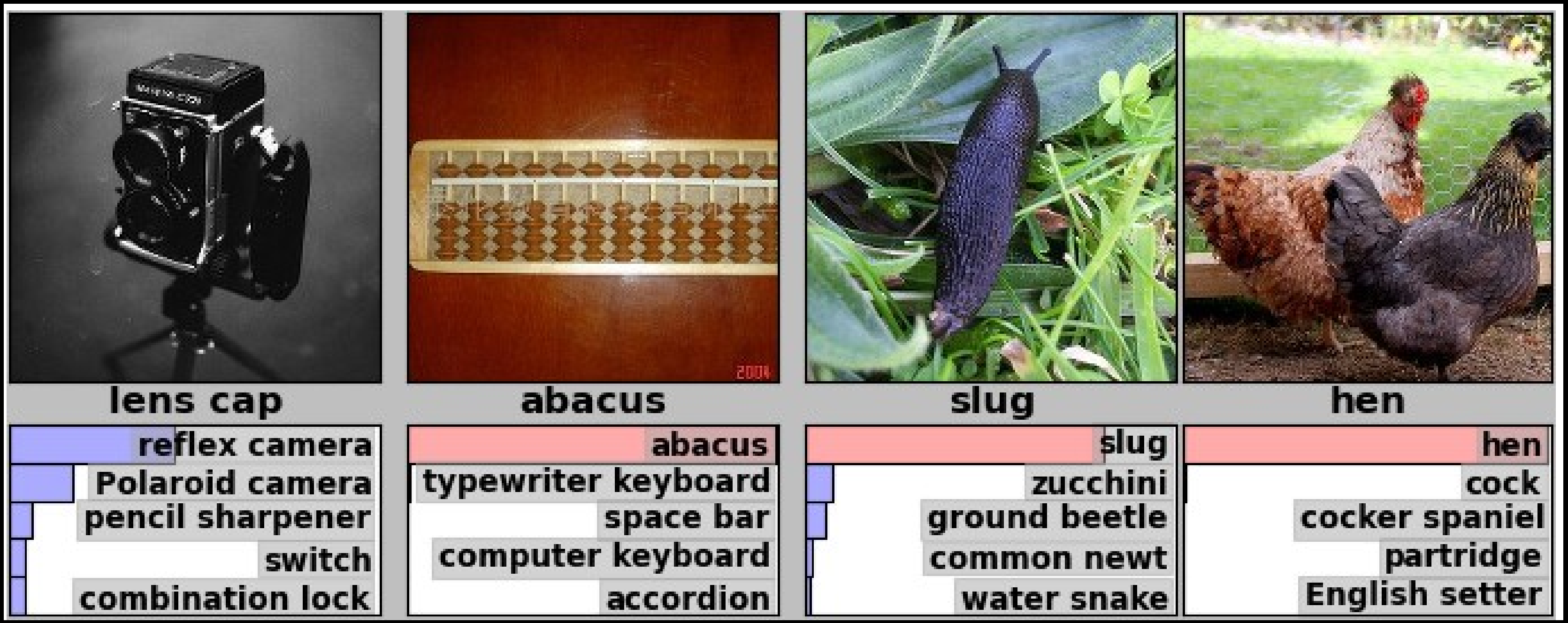
[The Return of] Convolutional Neural Networks

- Huge revival in 2012: Krizhevsky et al. NIPS 2012
- Still pretty much LeCun et al. 1989, just bigger models and larger training sets
- GPUs: nVidia Pascal 10 million times faster than 1980's Sun workstation



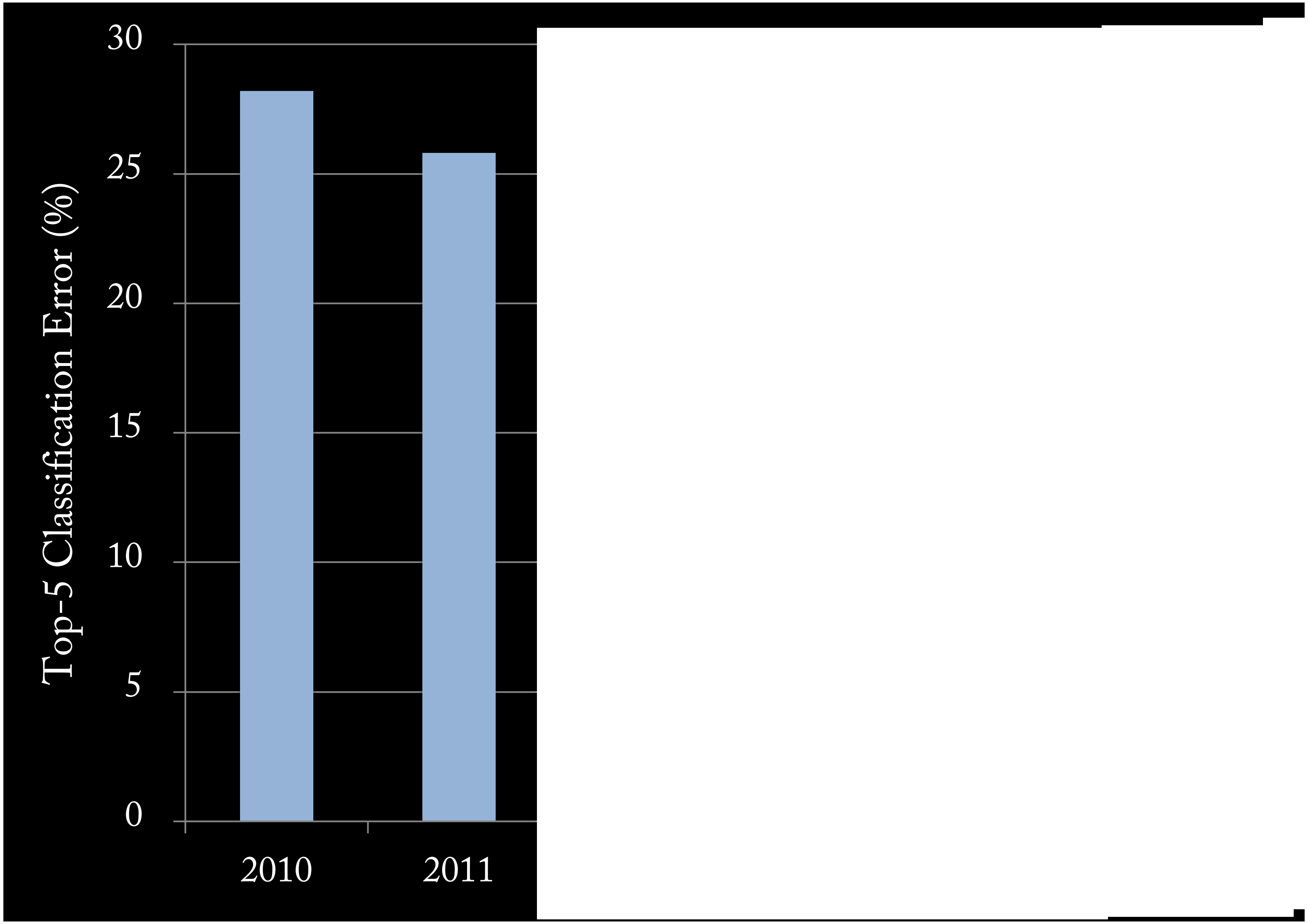
Object Recognition

- Image Classification
  – Pixels → Class Label

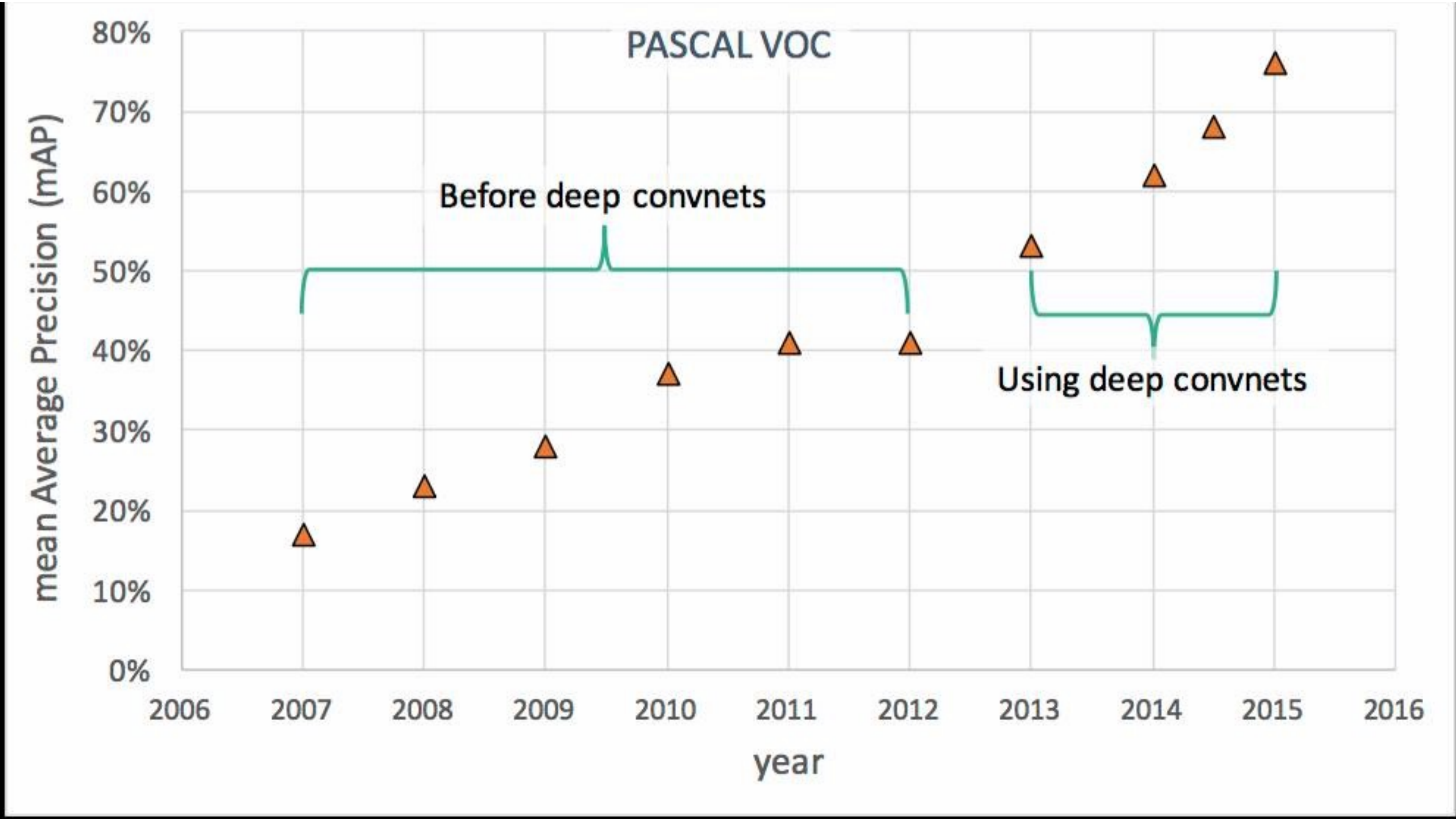


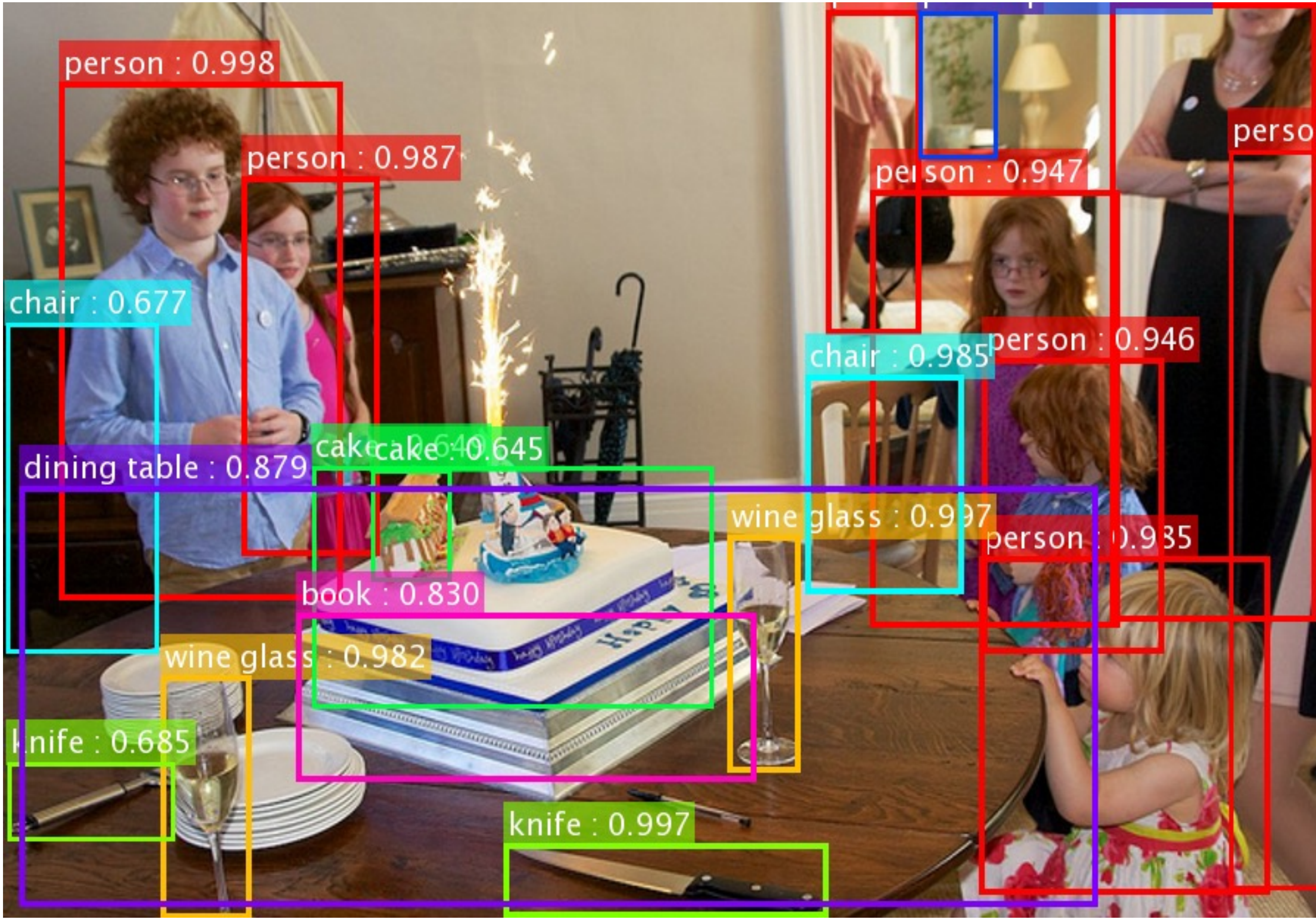
[Krizhevsky et al. NIPS 2012]

ImageNet Classification (2010 – 2015)

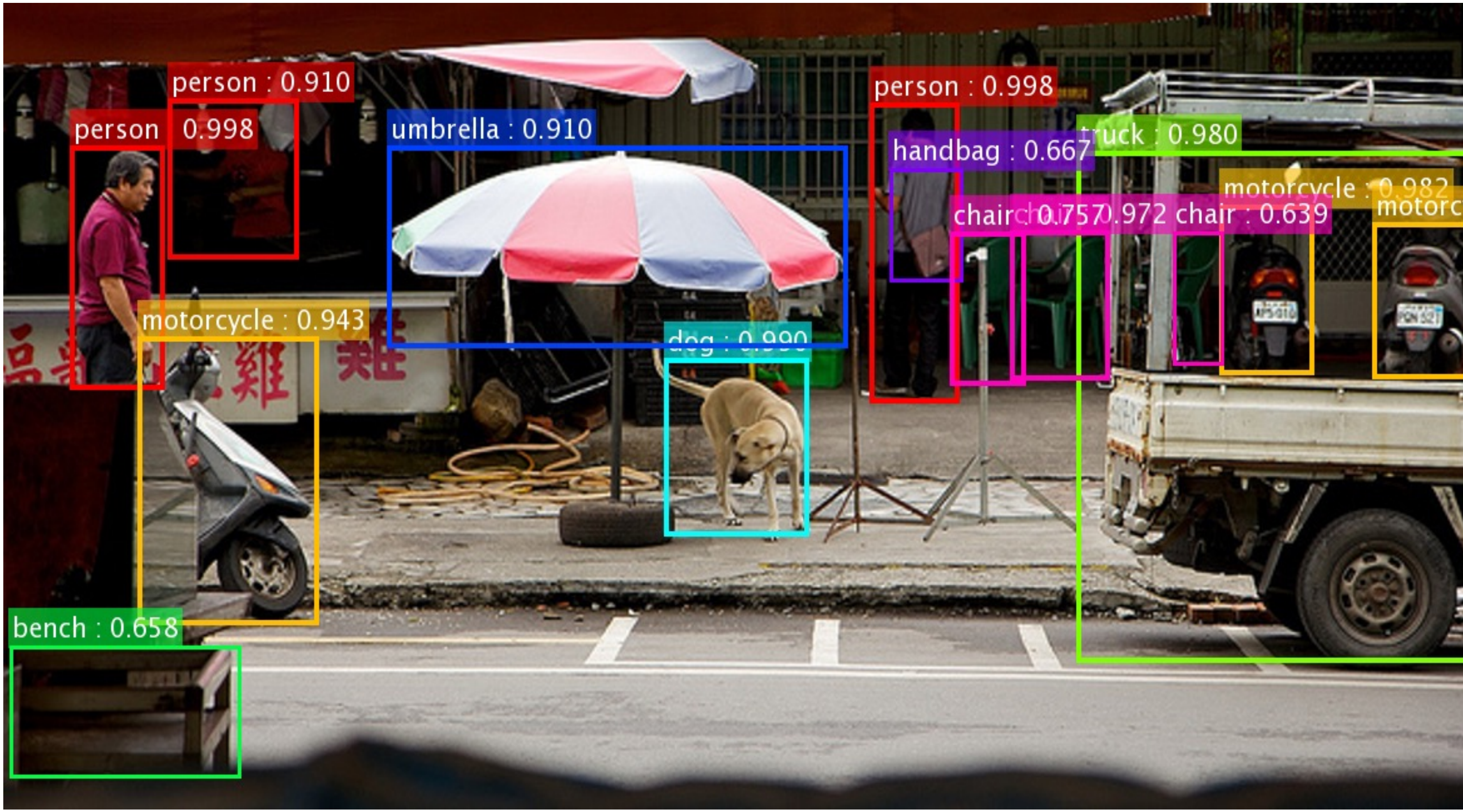


Object Detection Progress





He, Zhang, Ren, & Sun. “Deep Residual Learning for Image

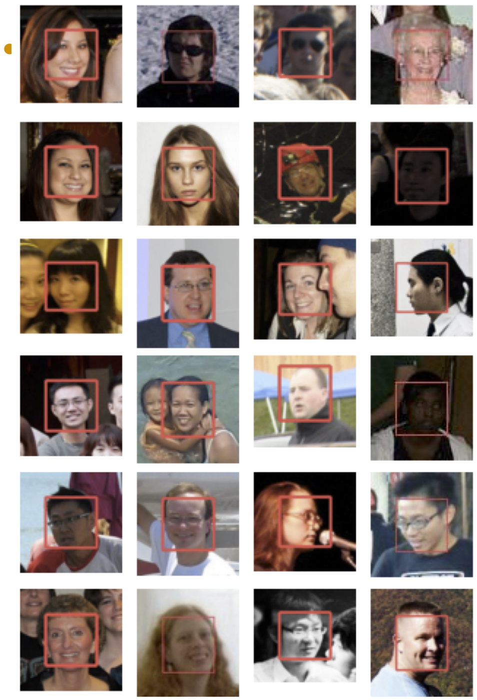


He, Zhang, Ren, & Sun. “Deep Residual Learning for Image

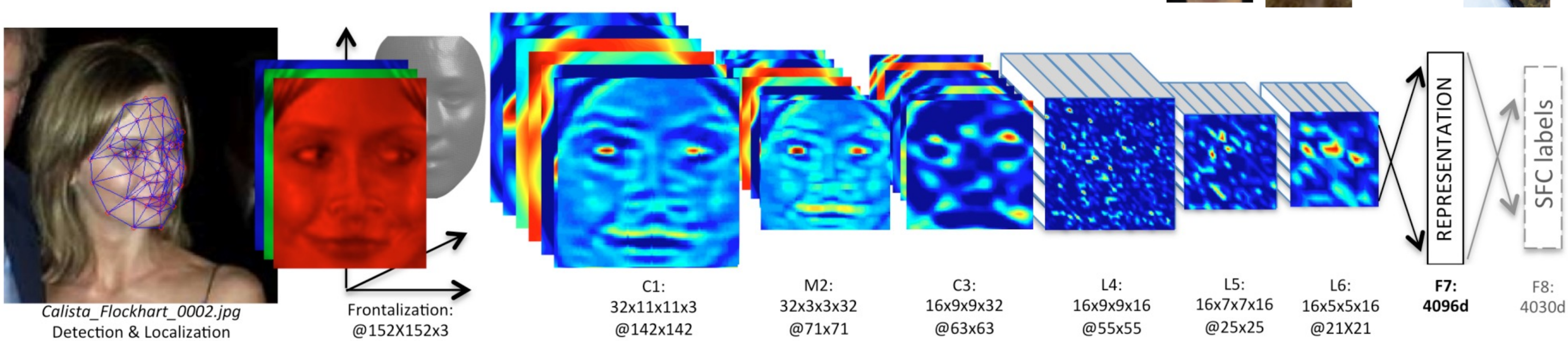
Face Detection (find faces)

- Real-time face detection on most phones/cameras now
- Use to set exposure
- Also input for face *recognition* system

Face Recognition (distinguish individuals)



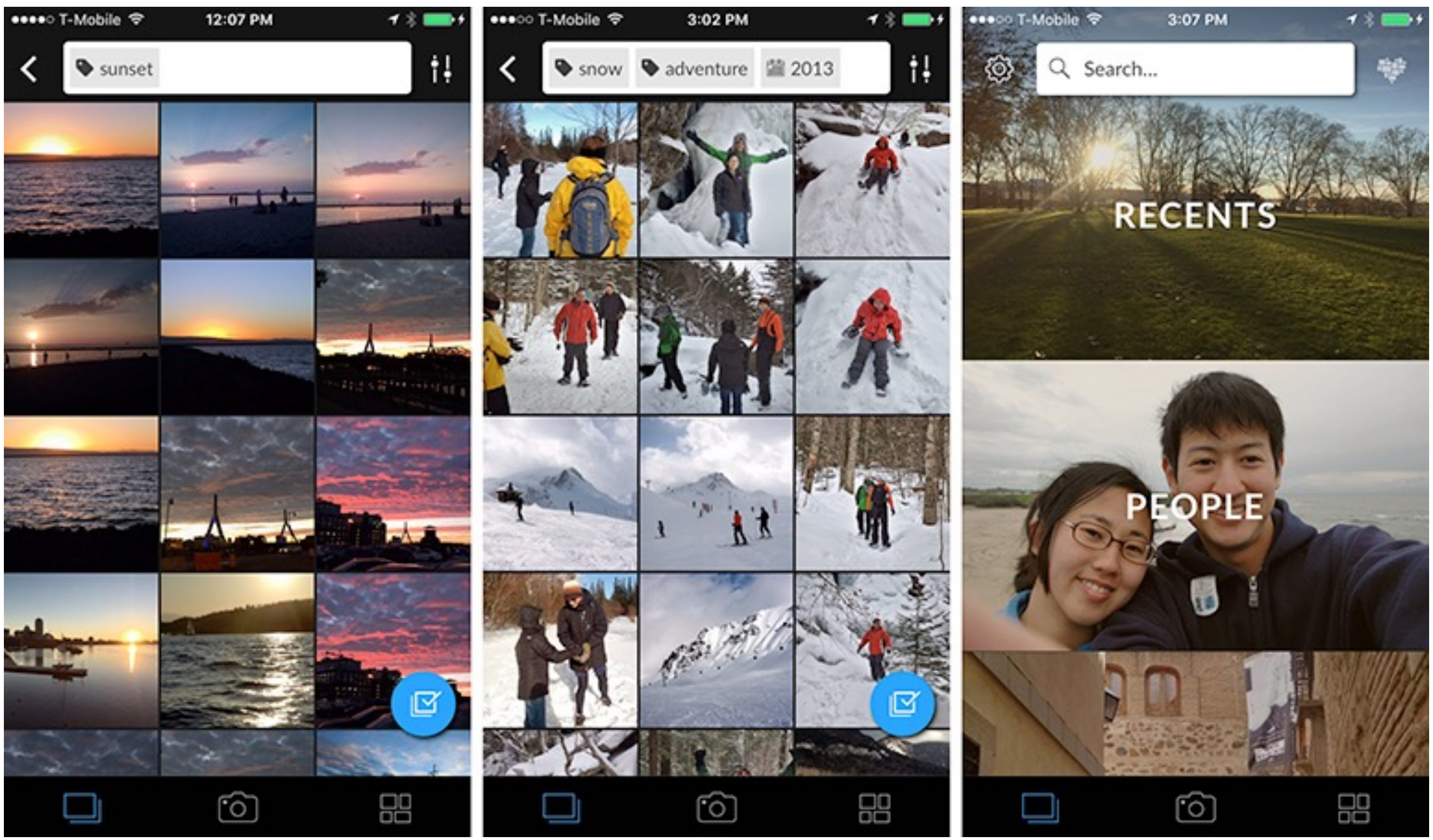
- Used by Facebook, Google etc.
- Tag people's faces in photos
- Need to distinguish a person's face from many others



[Taigman et al. DeepFace: Closing the Gap to Human-Level Performance in Face Verification, CVPR'14]

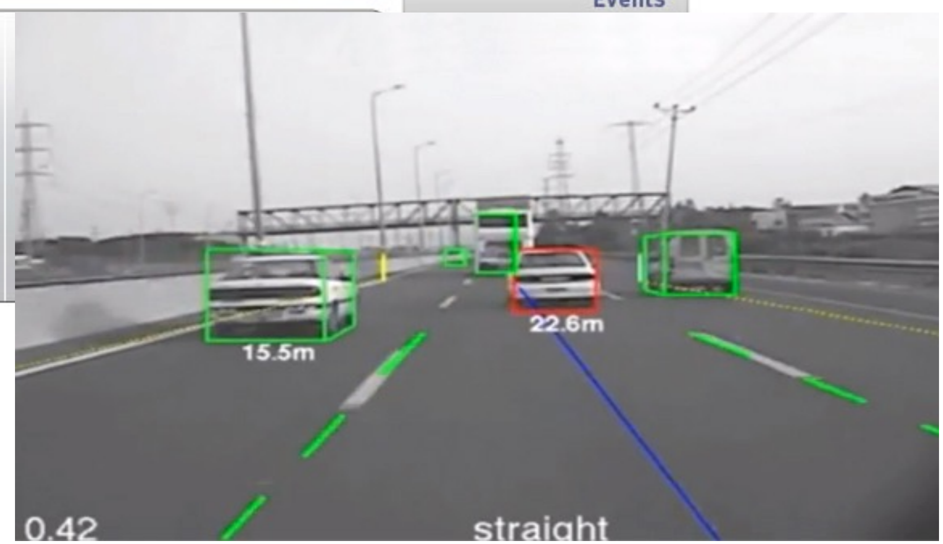
Advanced Photo Search

- Text-based image search
  - (that actually looks at image)



http://clarifai-blog.s3.amazonaws.com/wp-content/uploads/2015/12/pic1.jpg

Self-Driving Cars



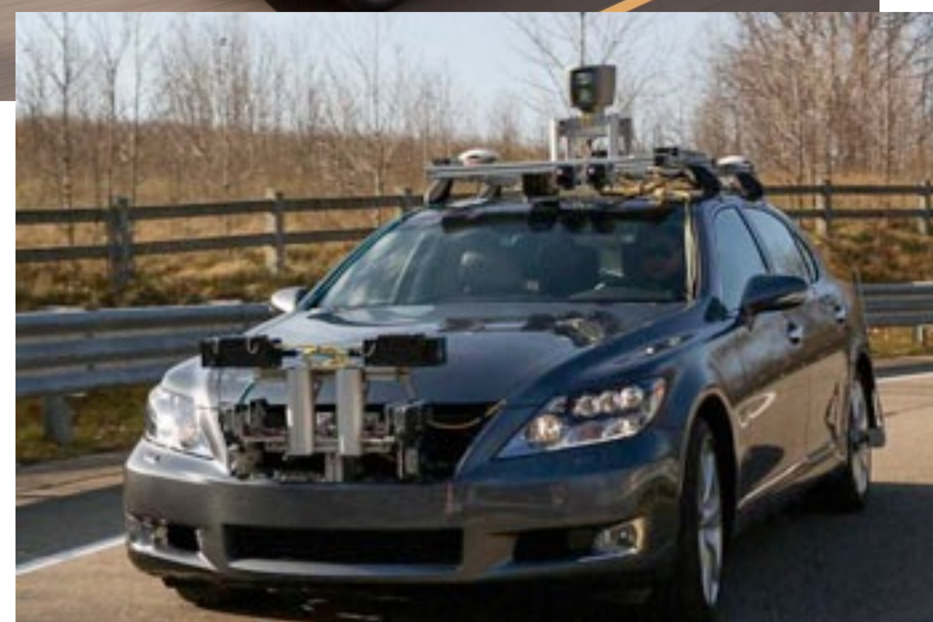
- Mobileye: Vision systems in high-end BMW, GM, Volvo models
- Very stringent accuracy requirements (not yet met)

Source: A. Shashua

Self-Driving Cars

- Many other companies:
  - Uber
  - Tesla
  - GM
  - Toyota

- More than just vision
  - LIDAR
  - Planning
  - Mapping
  - Anticipating behavior of other drivers

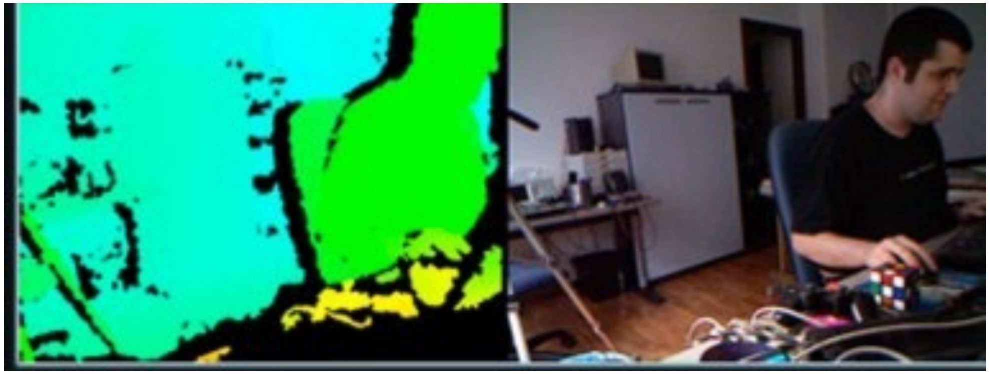


Virtual/Augmented Reality

- Tracking of user head w/high accuracy
- Rendering realistic 3D scene in real-time
- Oculus / HTC / Hololens

Vision-based interaction (and games)

Microsoft Kinect



Vision for robotics, space exploration

NASA'S Mars Exploration Rover Spirit captured this westward view from atop a low plateau where Spirit spent the closing months of 2007.

Vision systems (JPL) used for several tasks

- Panorama stitching
- 3D terrain modeling
- Obstacle detection, position tracking
- For more, read "Computer Vision on Mars" by Matthies et al.

Source: S. Seitz

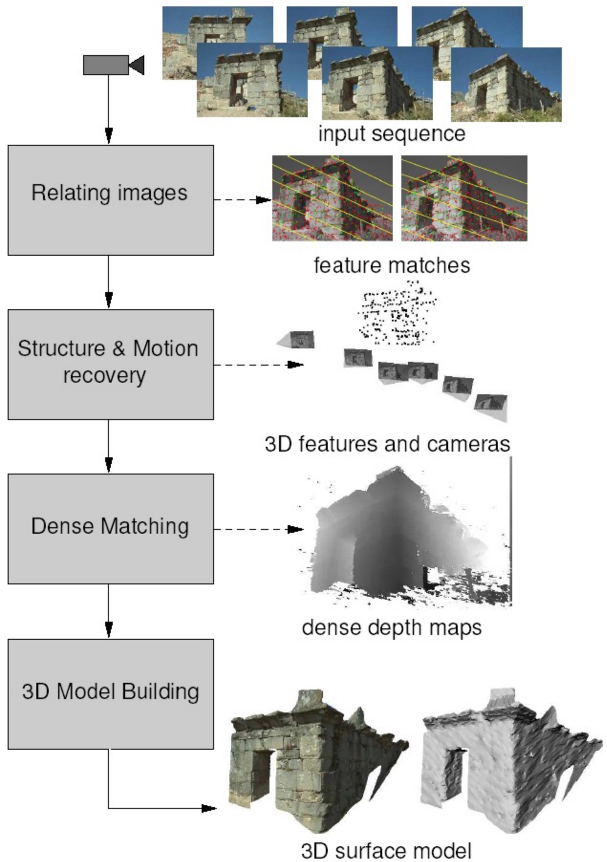
Novel view synthesis

Inputs: sparsely sampled images of scene

Outputs: *new* views of same scene

[NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis, Mildenhall et al. ECCV 2020]

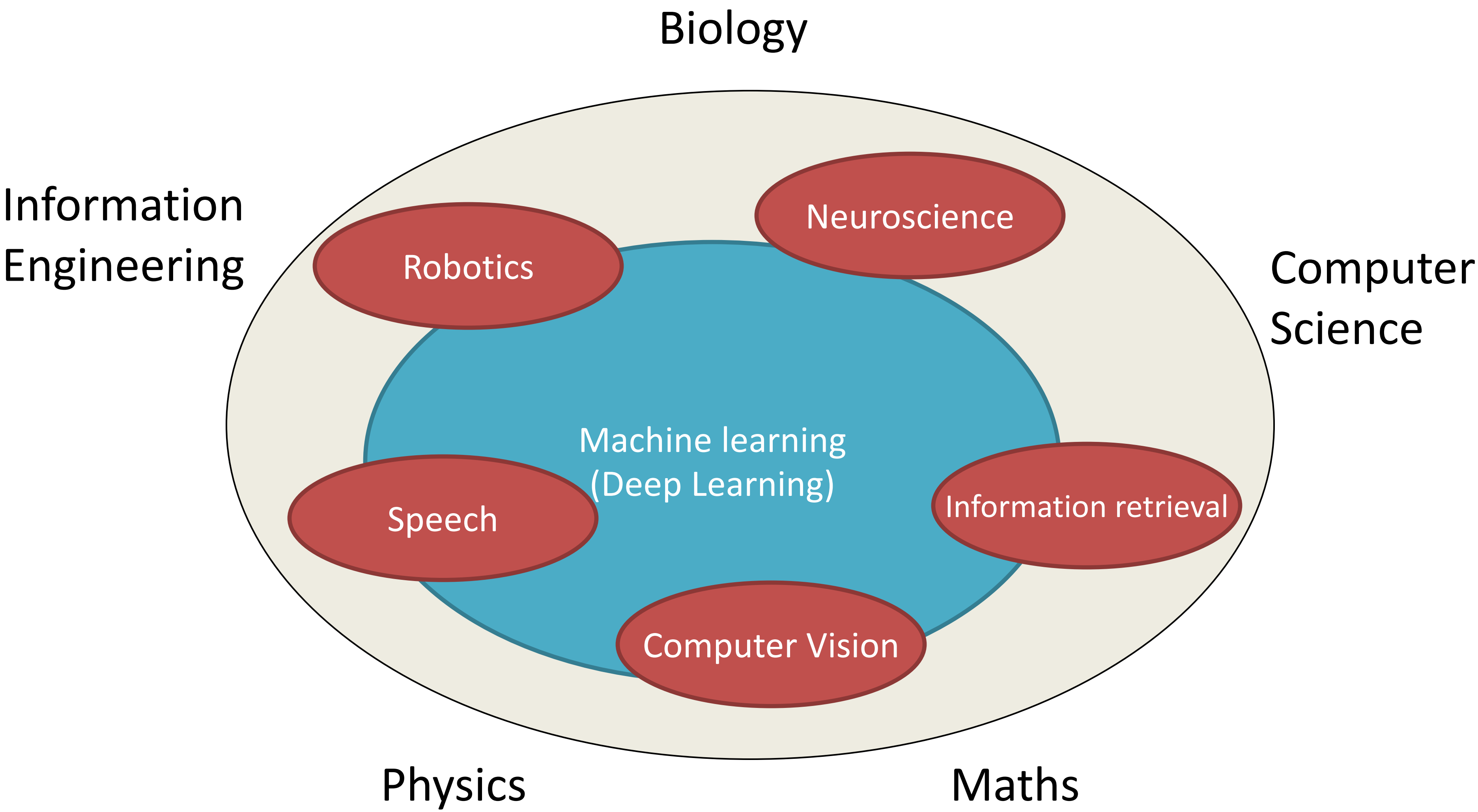
3D Reconstruction



Pollefeys et al.

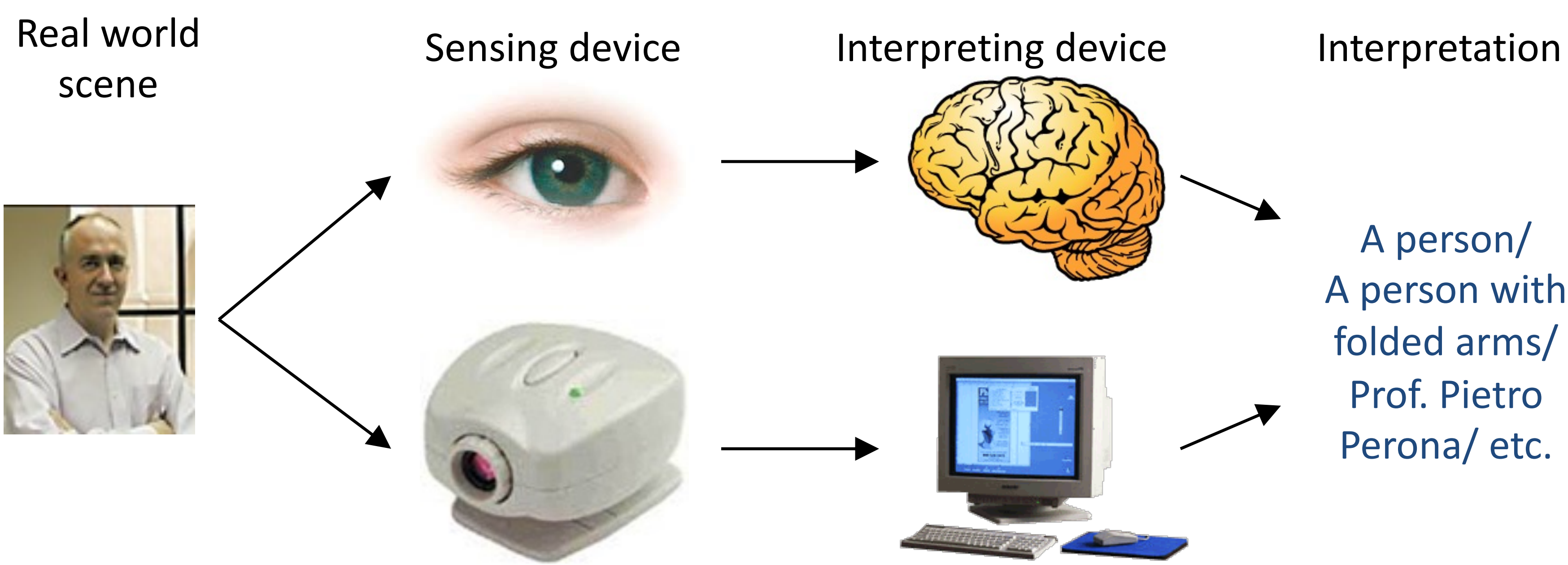
Goesele et al.

What is it related to?



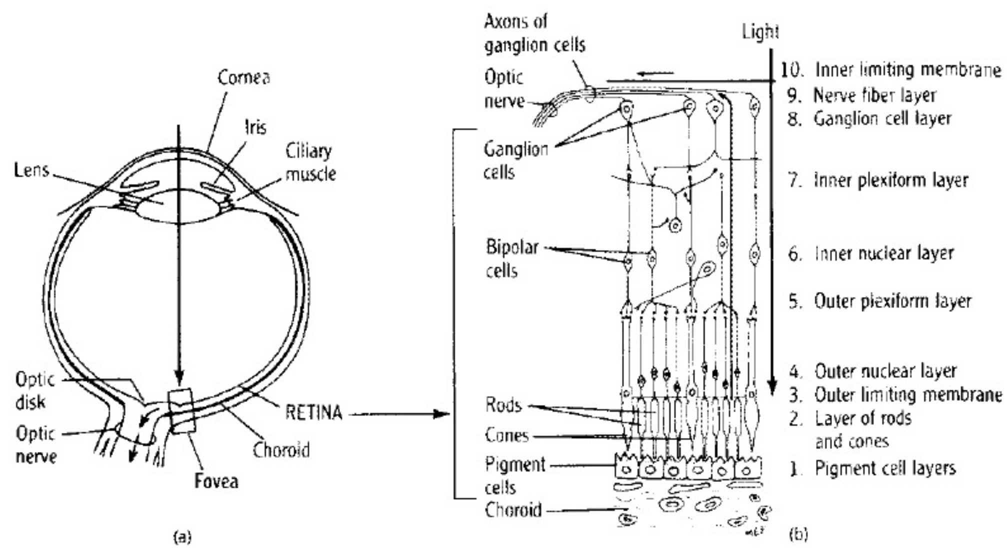
The problem

- Want to make a computer understand images
- We know it is possible – we do it effortlessly!



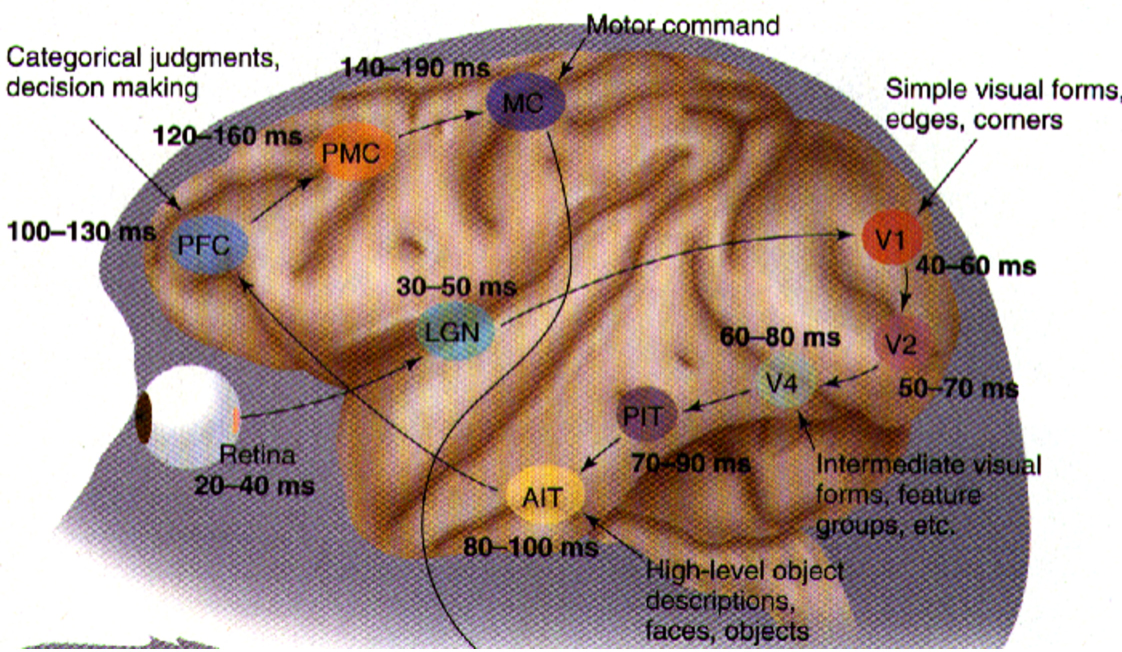
The Human Eye

- Retina measures about 5 × 5 cm and contains 10^8 sampling elements (rods and cones).
- The eye's spatial resolution is about 0.01° over a 150° field of view (not evenly spaced, there is a fovea and a peripheral region).
- Intensity resolution is about 11 bits/element, spectral range is 400–700nm.
- Temporal resolution is about 100 ms (10 Hz).
- Two eyes give a data rate of about 3 GBytes/s!



Human visual system

- Vision is the most powerful of our own senses.



[Thorpe et. al.]

- Around 1/3 of our brain is devoted to processing the signals from our eyes.

- The visual cortex has around $O(10^{11})$ neurons.

Vision as data reduction

- Raw feed from camera/eyes:
  - 10^{7-9} Bytes/s
- Extraction of edges and salient features
  - 10^{3-4} Bytes/s
- High-level interpretation of scene
  - 10^{1-2} Bytes/s

Why don't we just copy the human visual system?

- People try to but we don't yet have a sufficient understanding of how our visual system works.

- $O(10^{11})$ neurons used in vision

- By contrast, latest CPUs have $O(10^{8})$ transistors (most are cache memory)

- Very different architectures:
  - Brain is slow but parallel
  - Computer is fast but mainly serial

- Bird vs Airplane
  - Same underlying principles
  - Very different hardware

Admin Interlude

Course details

- Lecture recordings on Brightspace
- Course webpage:
  - http://cs.nyu.edu/~fergus/teaching/vision
- Piazza for discussions:
  - https://piazza.com/class/lm7352jomfb2k8
- Assignment submission
  - NYU Brightspace

Location

- 19 Washington Place, Room 102

- Office Hours
  - In person (+virtual): Thursday, 9pm onwards, i.e. right after class.
    - 19 Washington Place, Room 102

Class Teaching Assistants

Tutors:

- Rajeev Koppuravuri (rk4305@nyu.edu)
- Sriharsha Gaddipati (sg7372@nyu.edu)

Graders:

- Kranthi Kiran GV kranthi.gv@nyu.edu

Office hours to be announced (see website)

What you need

- Access to a computer than can run PyTorch
  - Open-source download
- GPU access:
  - Everyone should have been granted an NYU HPC account with Google Cloud access.
    - If not, please email me….
  - Class TAs will run a session showing you how to use. Please attend.

Pre-requisites

- Linear algebra
  - https://www.coursera.org/learn/linear-algebra-machine-learning
- Basic machine learning
  - E.g. Andrew Ng's Coursera course
- Coding in Python
  - PyTorch experience useful

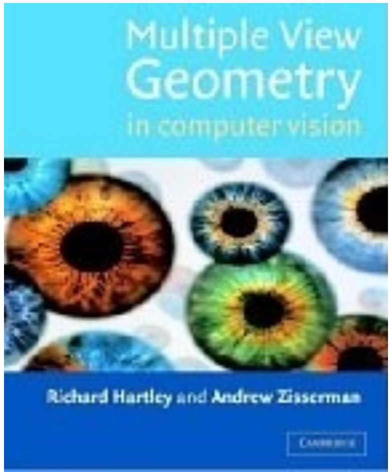
Textbooks

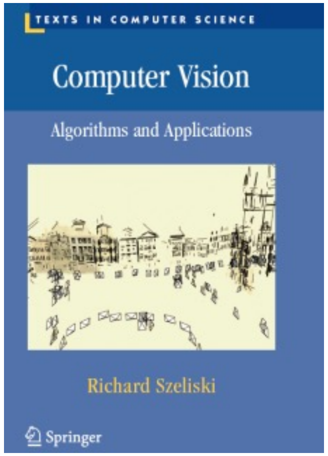
- Course does not use a textbook
- Deep Learning book (Goodfellow, Courville and Bengio)
  - http://www.deeplearningbook.org/
- Lots of pretty good blogs

- Geometric vision:

  Szeliski, R., Computer Vision. http://szeliski.org/Book/

  Hartley, R. and Zisserman, A. Multiple View Geometry in Computer Vision, Academic Press, 2002.





Grading

- Assignments (51%) + Course project (49%)
- Assignments on the course webpage are outdated: new ones will appear
- 3 assignments (51% of total)
  - 1^{st} = 17%. [Object classification]
  - 2^{nd} = 17% [Object Detection]
  - 3rd = 17% [3D computer vision]

Course Project

- Please choose by mid-October
  - Require project abstract
- Will put list of good project ideas up on Piazza
- Feel free to come up with your own!
  - Come to office hours to discuss
- Work in pairs (3 in a pinch)
  - Can use whatever platform you prefer
- Submit report + 2 min video instead of final exam. Due Monday December 18th.

Syllabus

- High-level vision
  - Introduction to neural nets
  - Convolutional nets (ConvNets)
  - Object recognition
  - Face recognition
  - Video recognition

- Low-level vision
  - Edge, corner, feature detection
  - Stereo reconstruction
  - Structure from motion, optical flow

- Other topics
  - Image processing tasks
  - Recurrent nets (images + text)
  - Generative models
  - Unsupervised learning

What the course will NOT cover

- Biology relating to vision
  - Go to CNS
- Huge detail on stereo reconstruction
  - Cool topic, but could easily be course of its own
- How to capture & enhance images
  - See Computational Photography course

Likely Deviations

- May have guest lecturers give some classes

End of
Admin Interlude

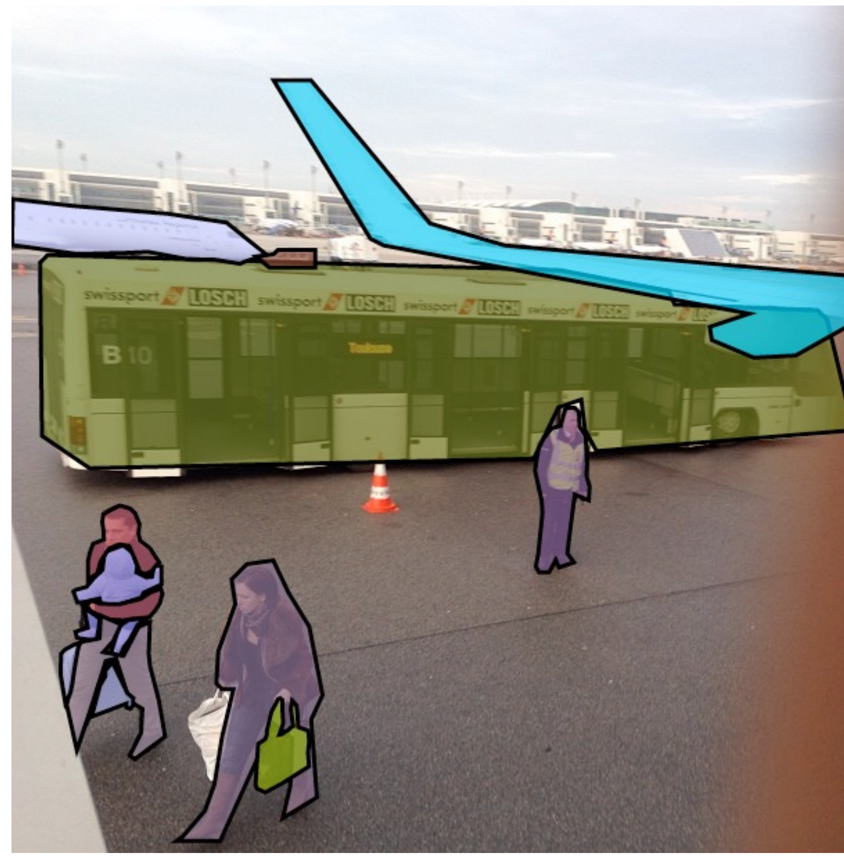
Computer Vision:
A whole series of problems

- What is in the image ?
  - Object recognition problem
- Where is it ?
  - 3D spatial layout
  - Shape
- How is the camera moving ?
- What is the action ?

Object Recognition

- "Understand objects in image"

- Different tasks:



Classification:

Image contains bus (binary yes/no)

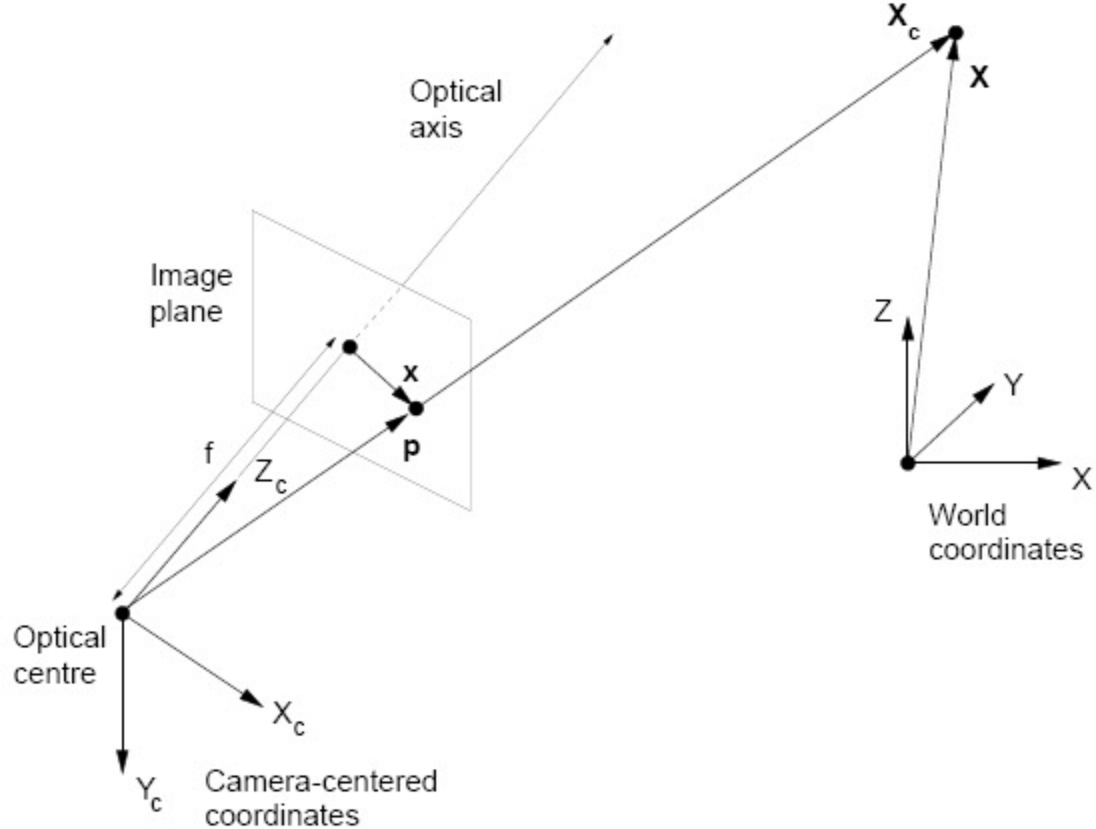
Detection:

Localize object instances
  (bounding box or mask)

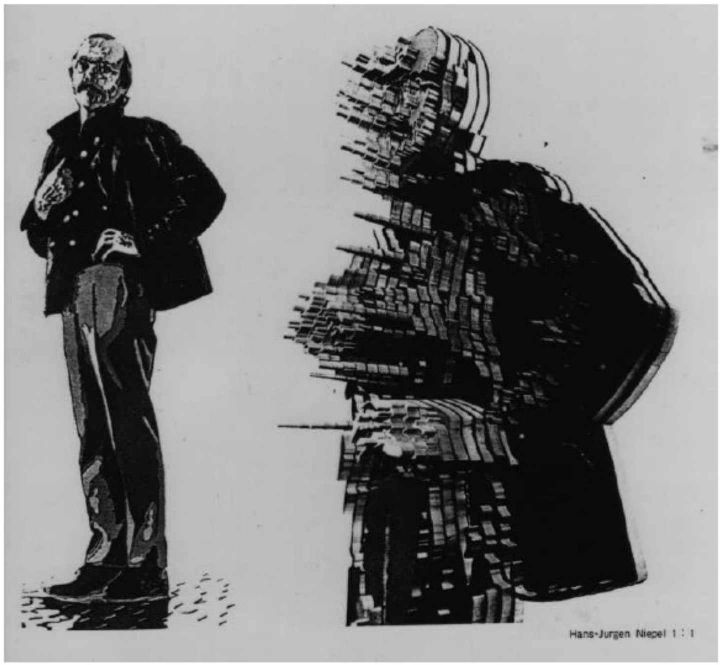
Semantic segmentation:

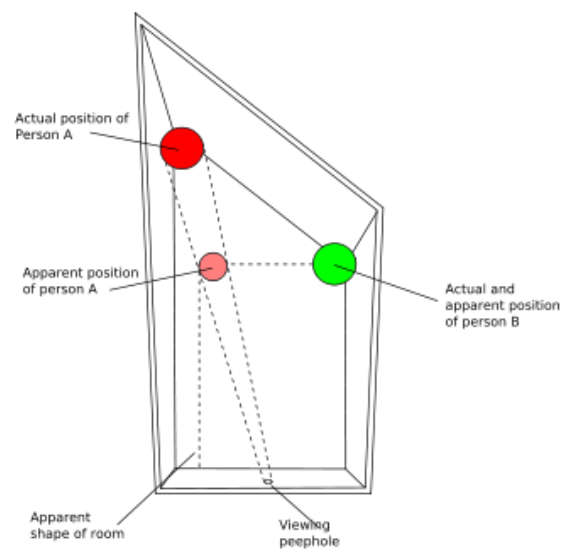
Label every pixel

Image is a projection of world



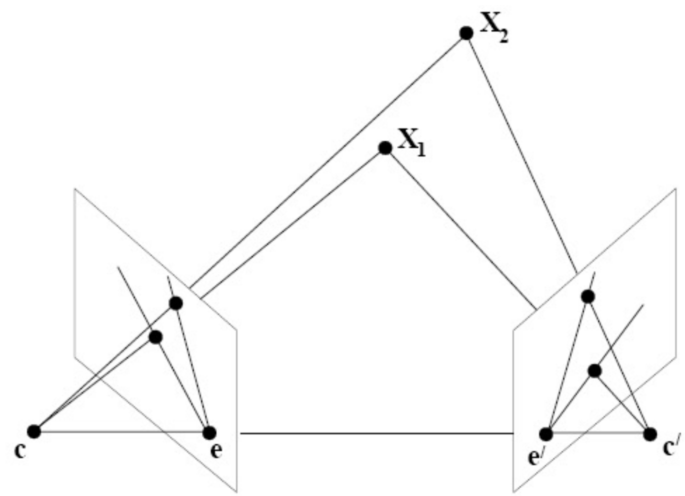
An under-constrained problem

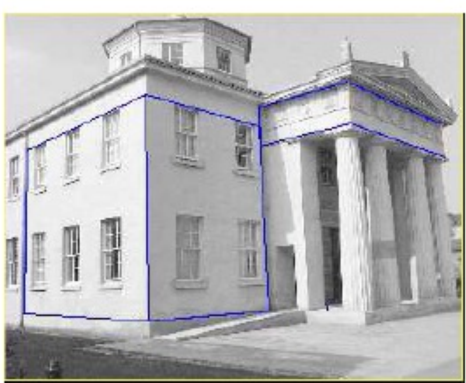




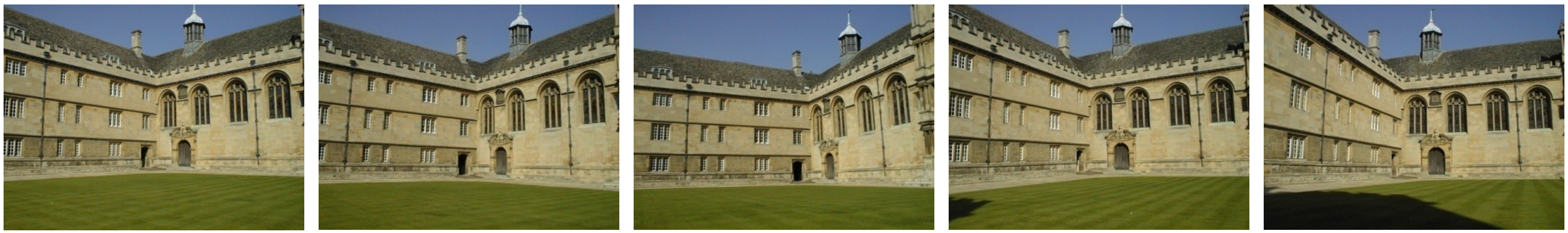
Stereo Vision

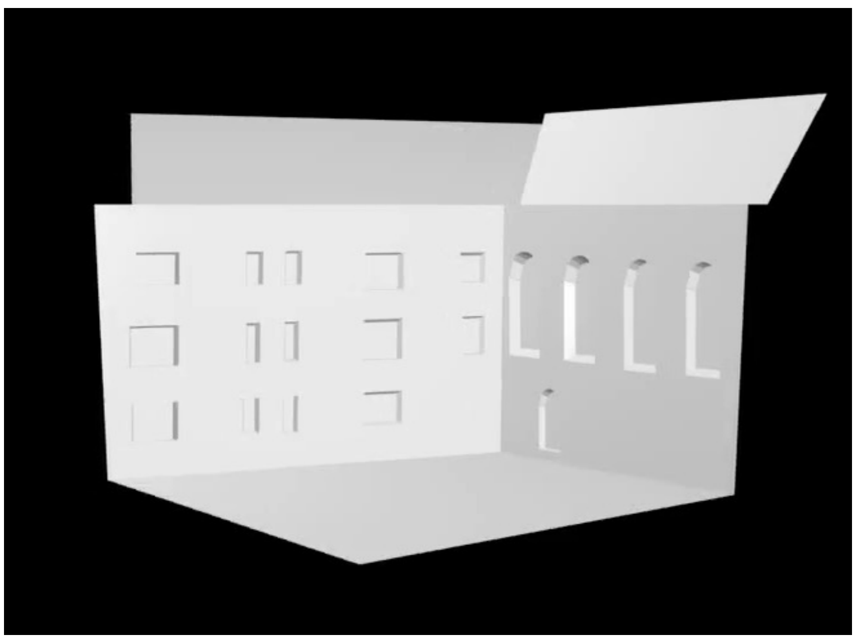
- By having two cameras, we can triangulate features in the left and right images to obtain depth.
- Need to match features between the two images:
  - Correspondence Problem

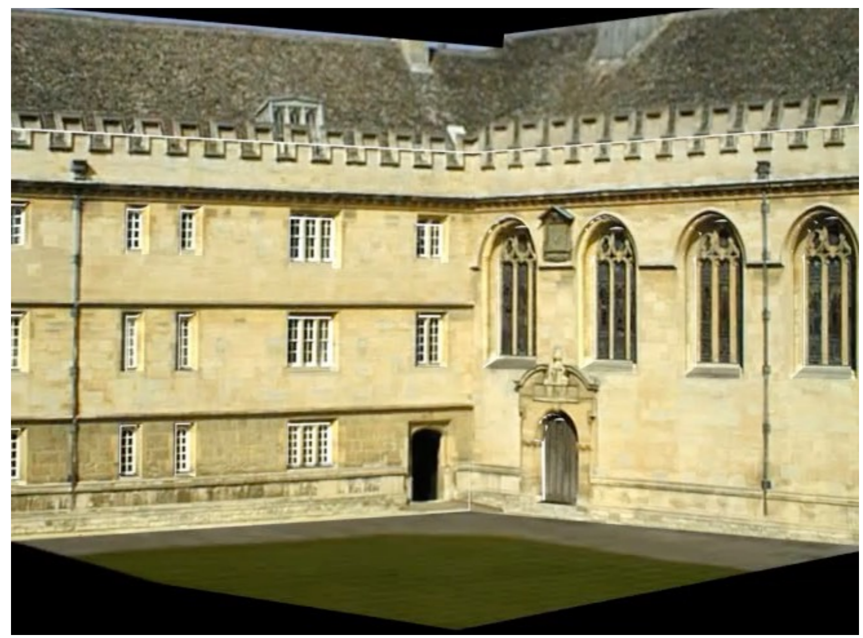




Geometry:
3D models of planar objects







[Fitzgibbon et. al]
[Zisserman et. al.]

Structure and Motion Estimation

Objective: given a set of images …

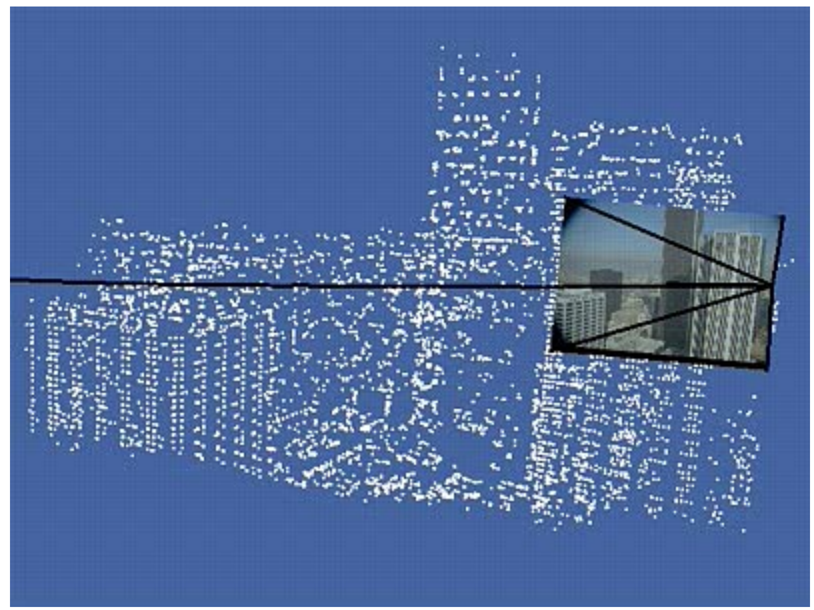
Want to compute where the camera is for each image and the 3D scene structure:

- Uncalibrated cameras
- Automatic estimation from images (no manual clicking)

Example

Image sequence

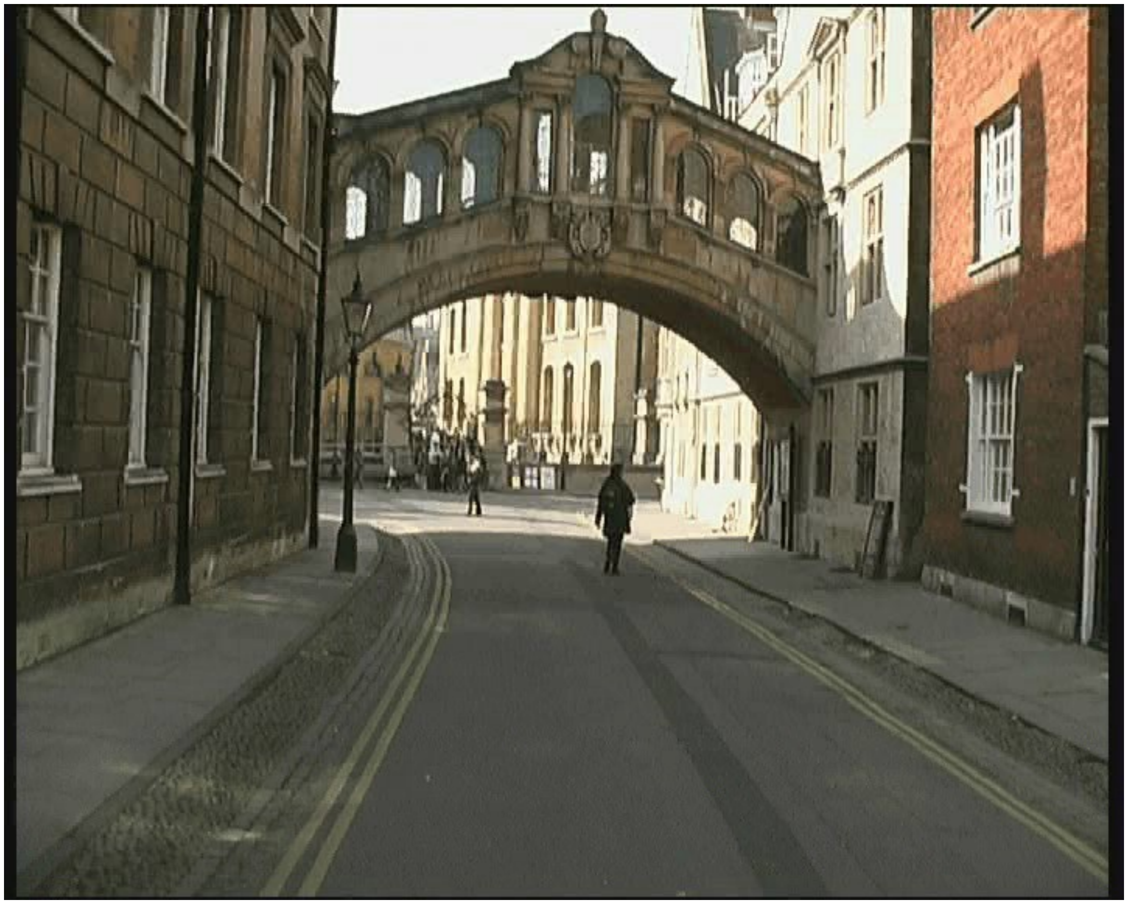
Camera path and points



[Fitzgibbon et. al]
[et. al. Zisserman]

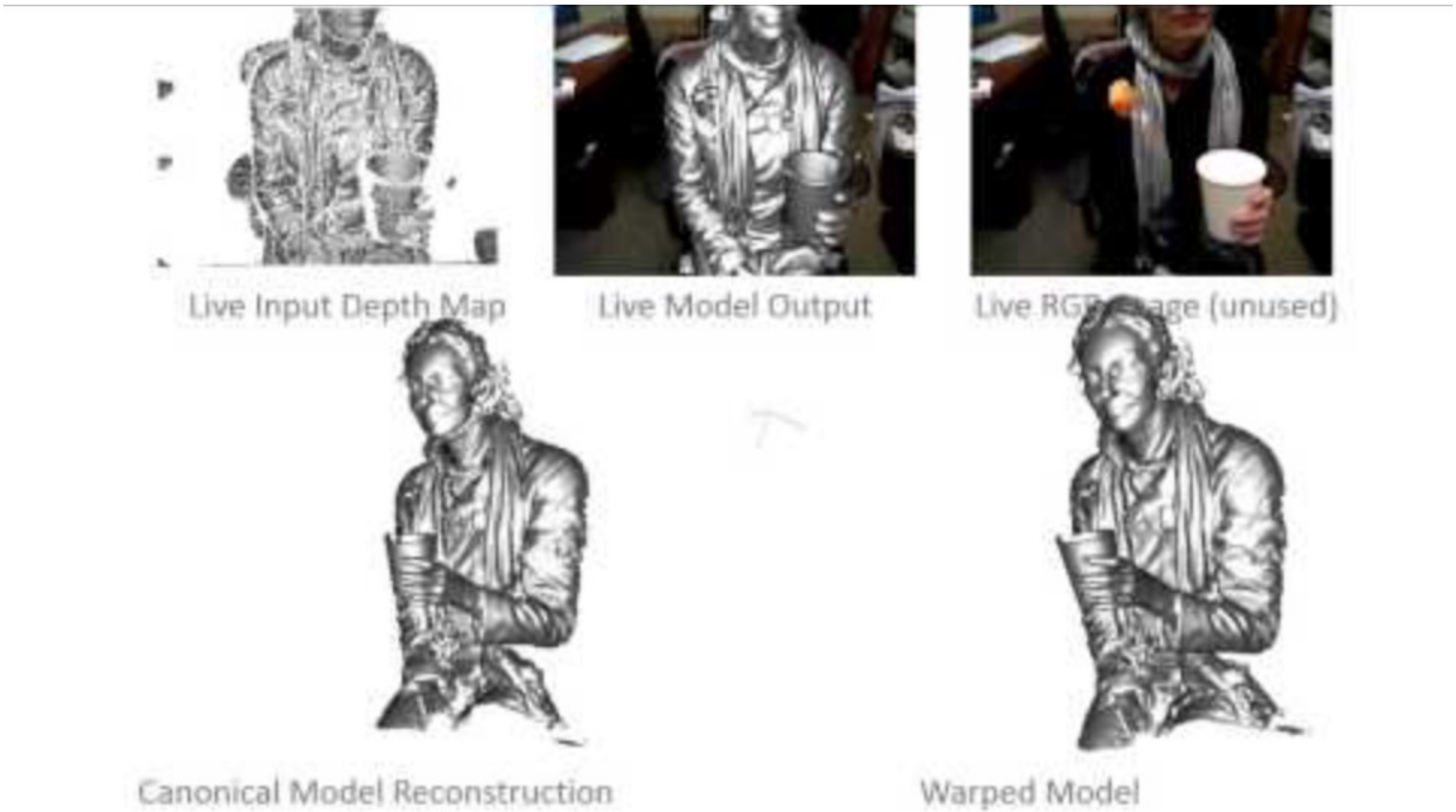
Application: Augmented reality

original sequence



Augmented

DynamicFusion

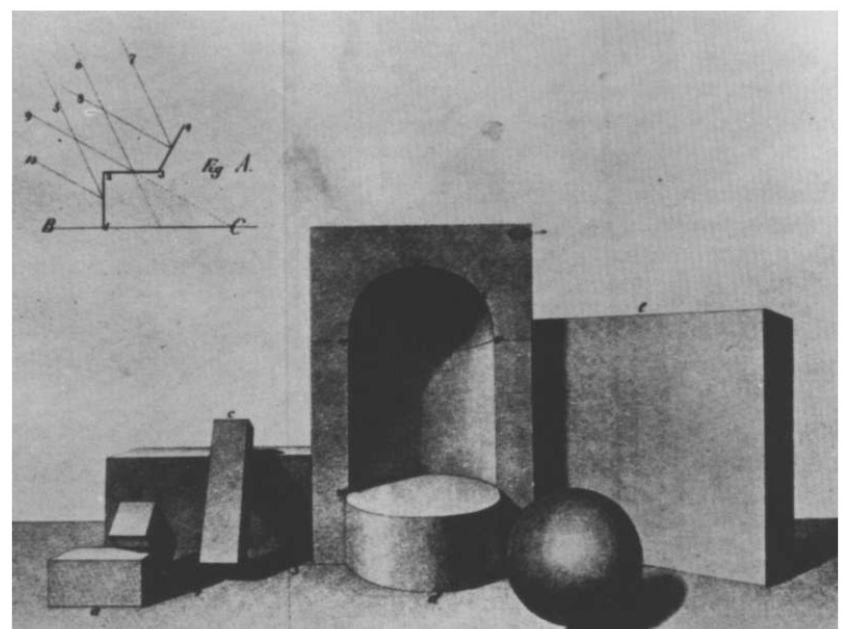


https://www.youtube.com/watch?v=i1eZekcc_IM

Interpretation from limited cues

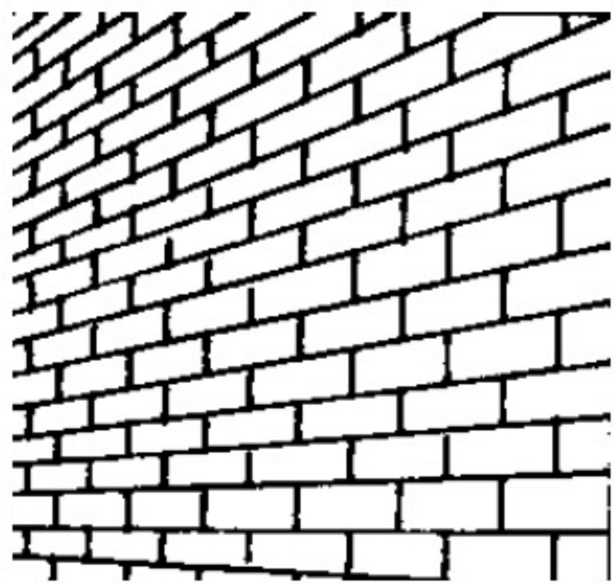
Shape from Shading

- Recover scene structure from shading in the image
- Typically need to assume:
  - Lambertian lighting, isotropic reflectance



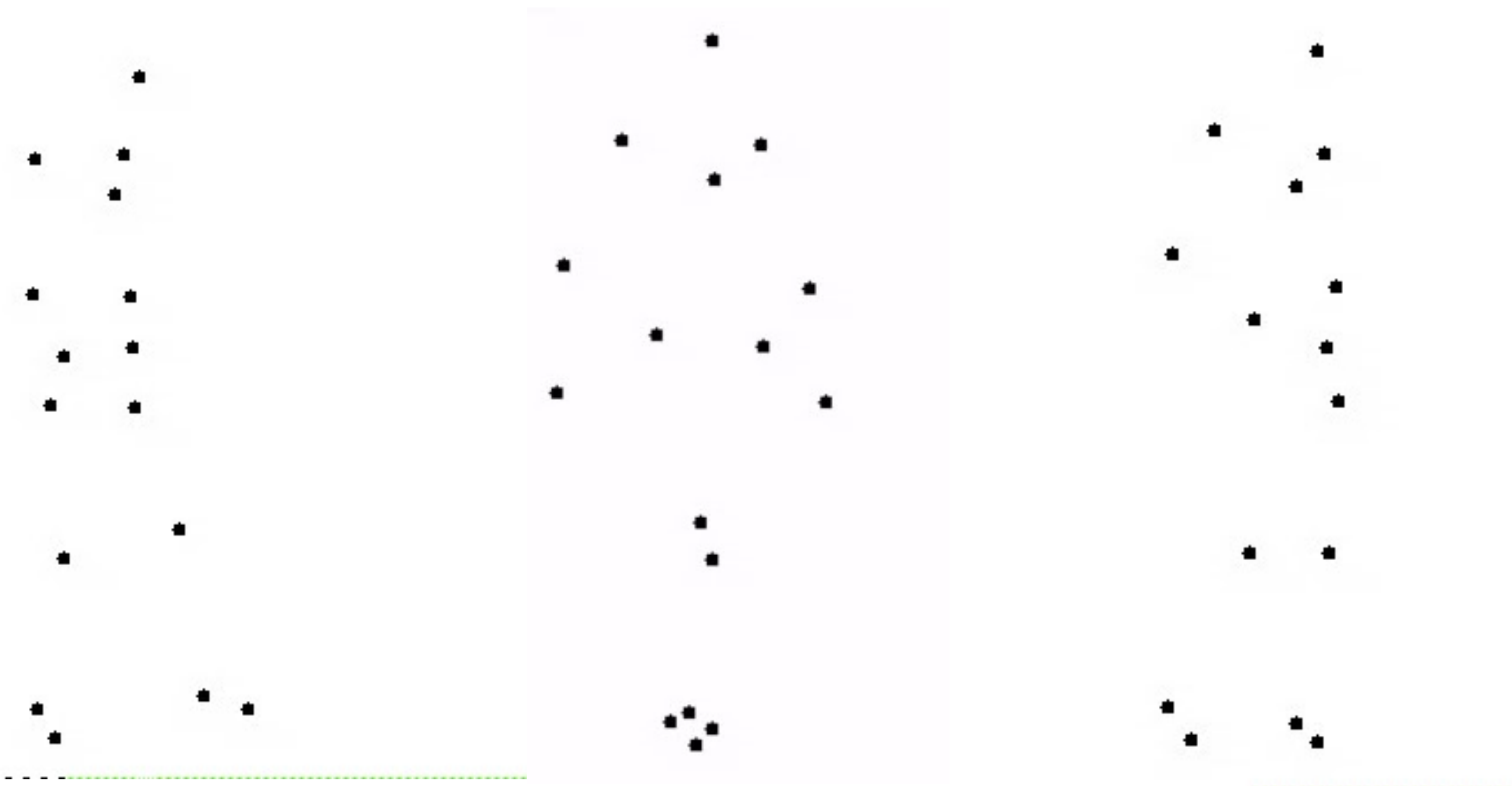
Shape from Texture

- Texture provides a very strong cue for inferring surface orientation in a single image.
- Necessary to assume homogeneous or isotropic texture.
- Then, it is possible to infer the orientation of surfaces by analyzing how the texture statistics vary over the image.



Human motion detection

Johansson's experiments ['70s]



Can you tell what it is?

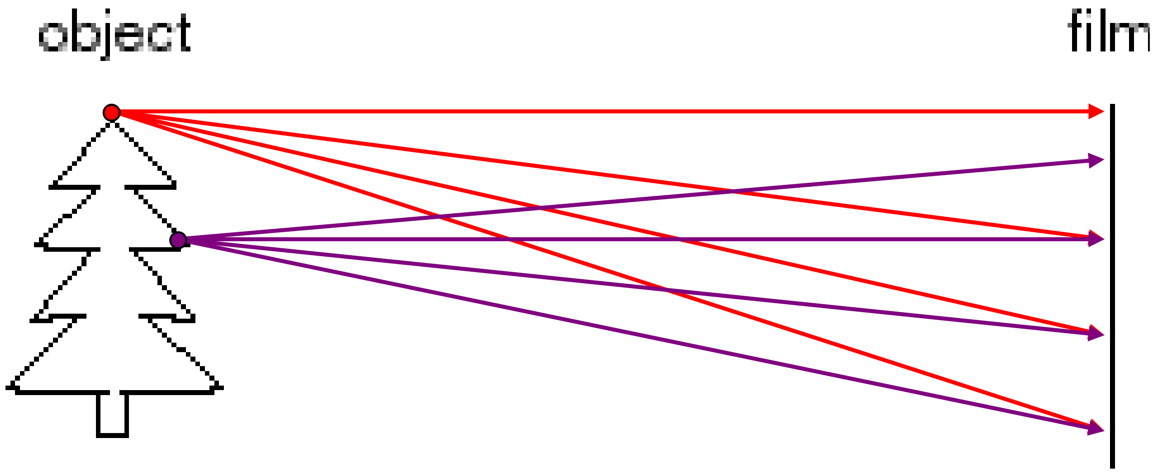
Cameras & Image Formation

Slides from: F. Durand, S. Seitz, S. Lazebnik, S. Palmer

Overview

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

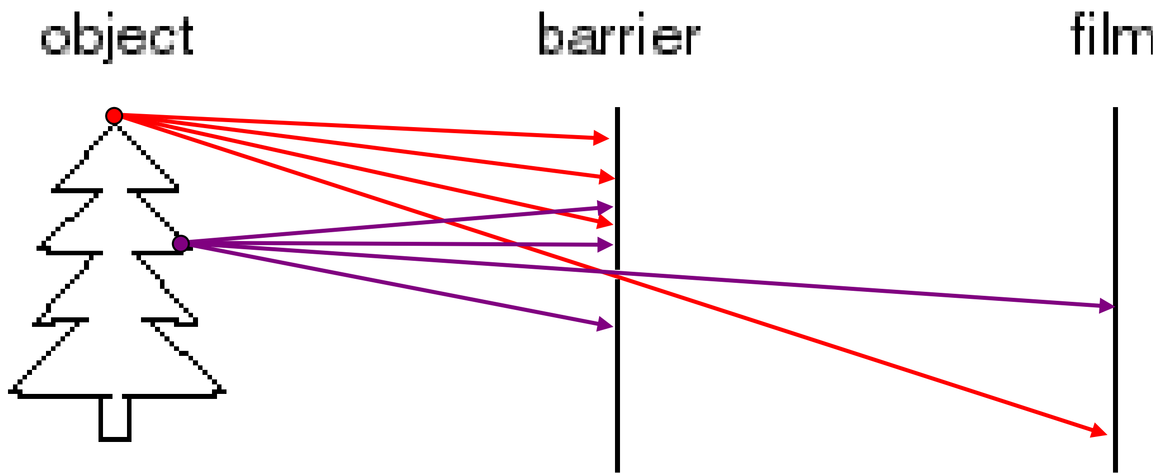
Let's design a camera



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

Slide by Steve Seitz

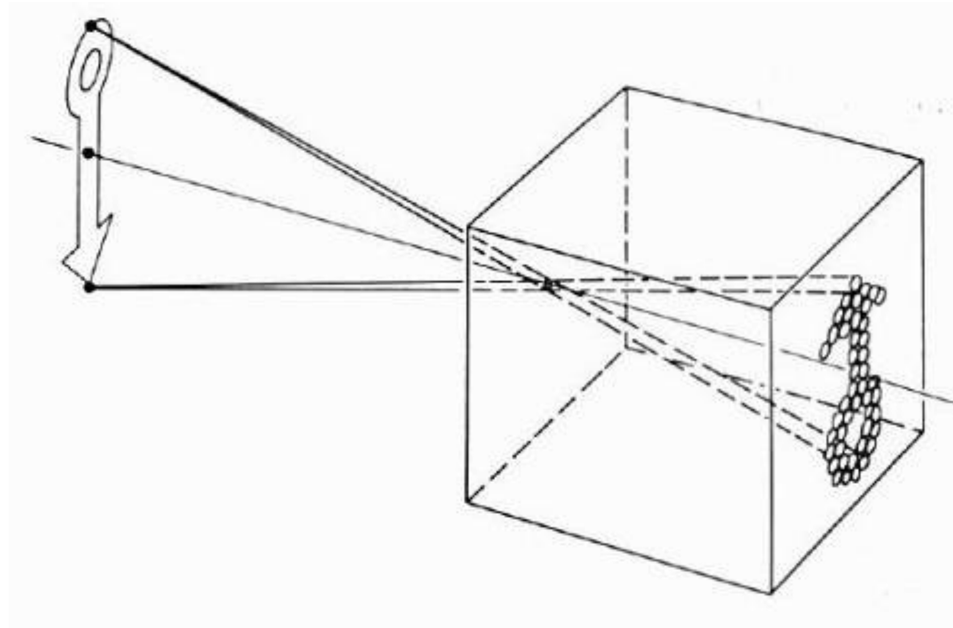
Pinhole camera



- Add a barrier to block off most of the rays
  - This reduces blurring
  - The opening is known as the **aperture**

Slide by Steve Seitz

Pinhole camera model

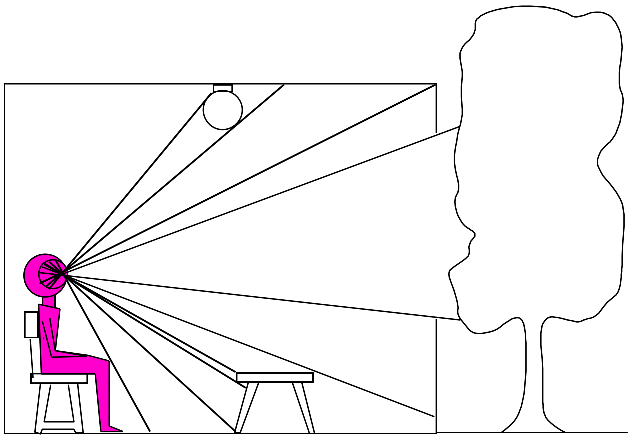


- Pinhole model:
  - Captures **pencil of rays** – all rays through a single point
  - The point is called **Center of Projection (focal point)**
  - The image is formed on the **Image Plane**

Slide by Steve Seitz

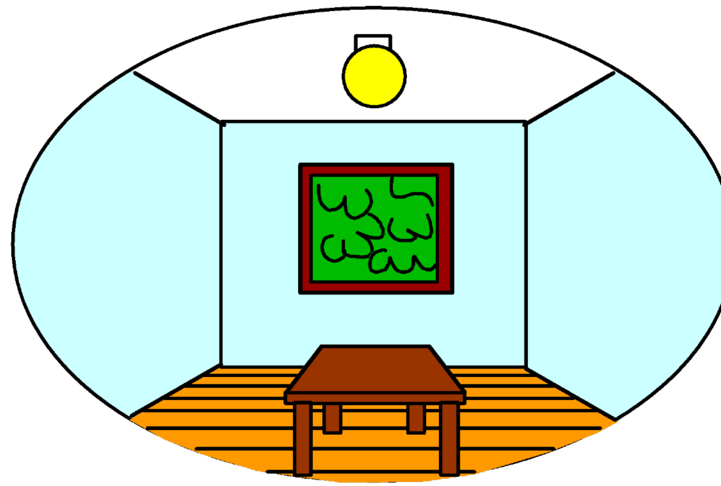
Dimensionality Reduction Machine (3D to 2D)

3D world



Point of observation

2D image



What have we lost?

- Angles
- Distances (lengths)

Slide by A. Efros

Projection properties

- Many-to-one: any points along same *visual ray* map to same point in image
- Points → points
  - But projection of points on focal plane is undefined
- Lines → lines (collinearity is preserved)
  - But line through focal point (visual ray) projects to a point
- Planes → planes (or half-planes)
  - But plane through focal point projects to line

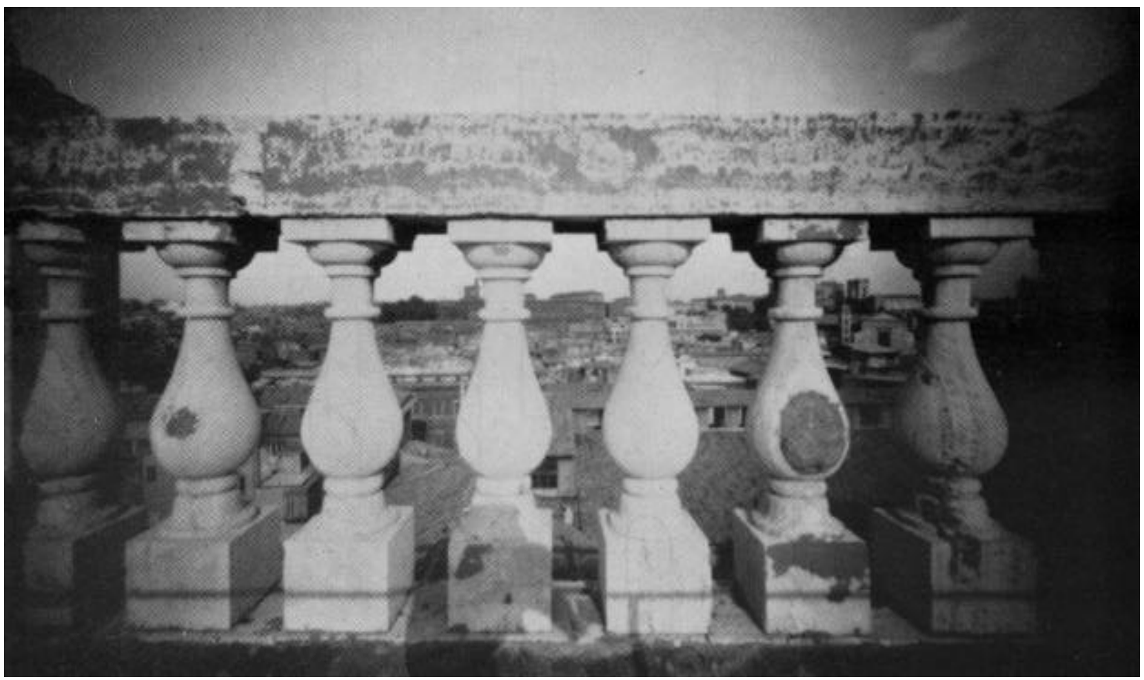
Perspective distortion

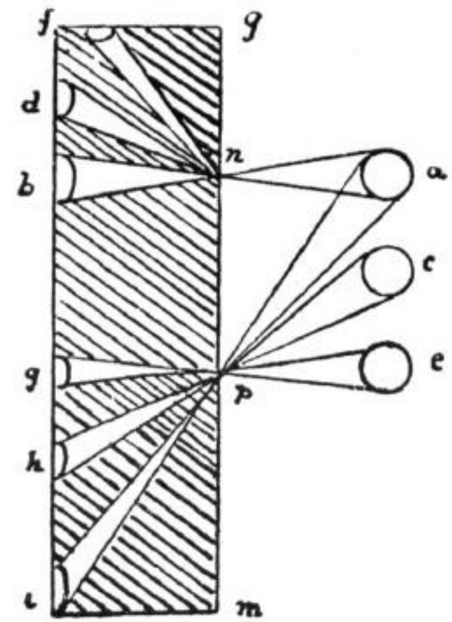
- Problem for architectural photography: converging verticals

Source: F. Durand

Perspective distortion

- The exterior columns appear bigger
- The distortion is not due to lens flaws

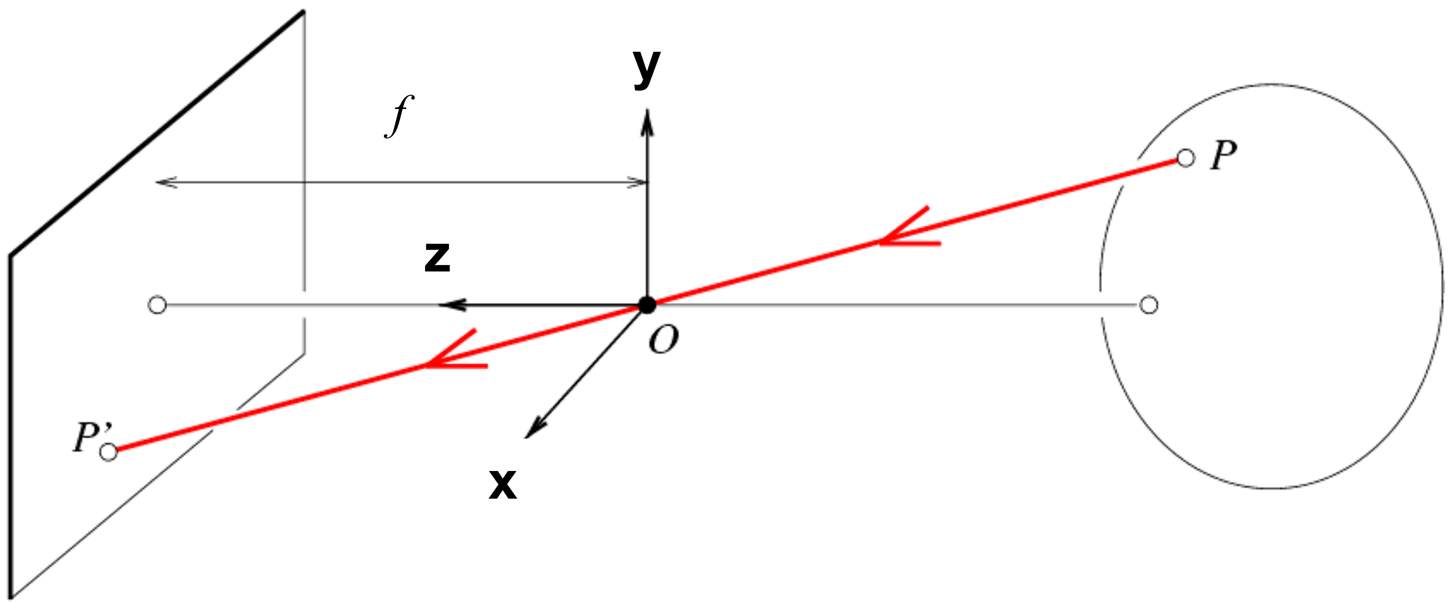




Slide by F. Durand

Perspective distortion: People

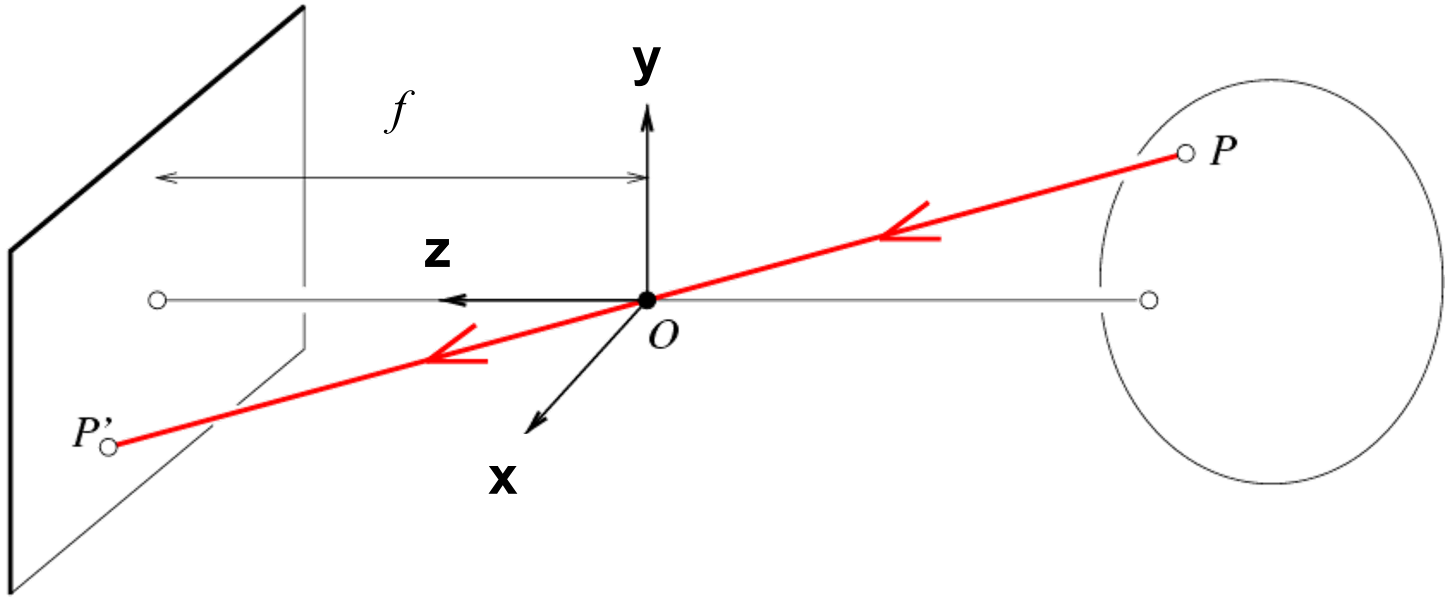
Modeling projection



- The coordinate system
  - The optical center (***O***) (aka focal point / center of projection) is at the origin
  - Optical axis is in z direction
  - The image plane is parallel to xy-plane (perpendicular to z axis)

Source: J. Ponce, S. Seitz

Modeling projection



- Projection equations
  - Compute intersection with image plane of ray from ***P*** = (x,y,z) to ***O***
  - Derived using similar triangles

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

- We get the projection by throwing out the last coordinate:

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

Source: J. Ponce, S. Seitz

Homogeneous coordinates

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

- Is this a linear transformation?
  - no—division by z is nonlinear

Trick: add one more coordinate:

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

homogeneous image coordinates

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

homogeneous scene coordinates

Converting *from* homogeneous coordinates

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

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

Slide by Steve Seitz

Perspective Projection Matrix

- Projection is a matrix multiplication using homogeneous coordinates:

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

divide by the third coordinate

Perspective Projection Matrix

- Projection is a matrix multiplication using homogeneous coordinates:

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

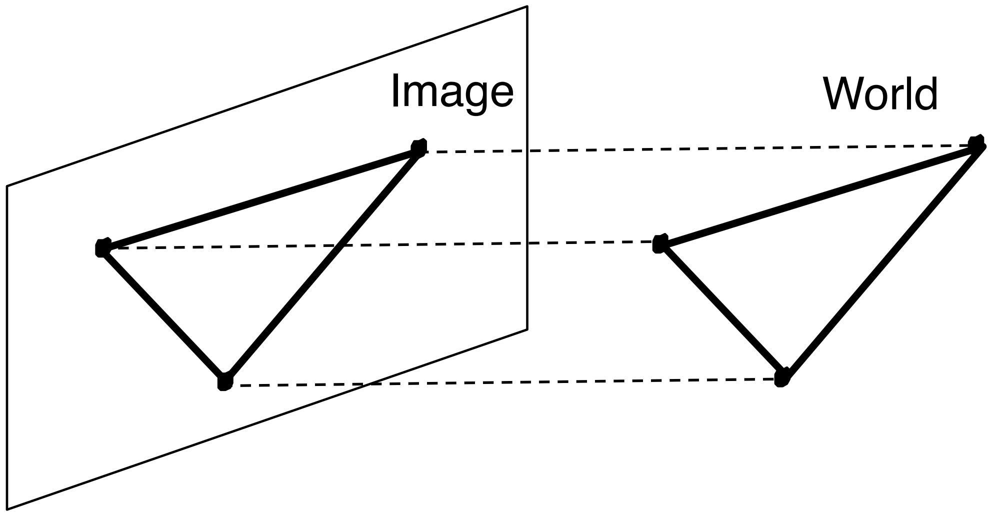
divide by the third coordinate

In practice: split into lots of different coordinate transformations…

$$\begin{pmatrix} \text{2D point} \\ \text{(3x1)} \end{pmatrix} = \begin{pmatrix} \text{Camera to pixel coord.} \\ \text{trans. matrix} \\ \text{(3x3)} \end{pmatrix} \begin{pmatrix} \text{Perspective} \\ \text{projection matrix} \\ \text{(3x4)} \end{pmatrix} \begin{pmatrix} \text{World to camera coord.} \\ \text{trans. matrix} \\ \text{(4x4)} \end{pmatrix} \begin{pmatrix} \text{3D point} \\ \text{(4x1)} \end{pmatrix}$$

Orthographic Projection

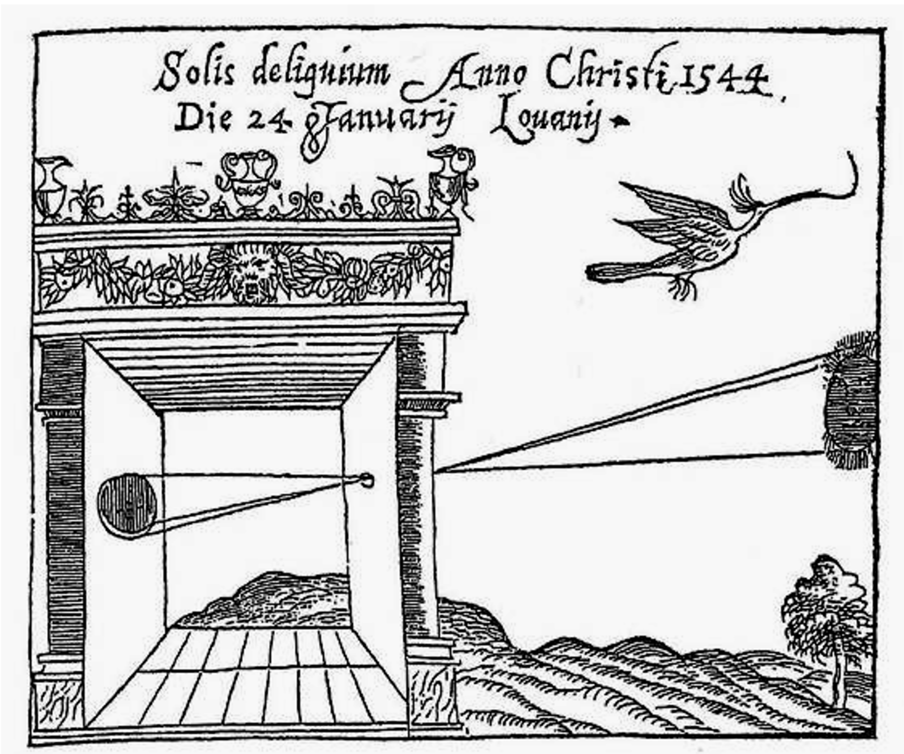
- Special case of perspective projection
  - Distance from center of projection to image plane is infinite



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

Building a real camera

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

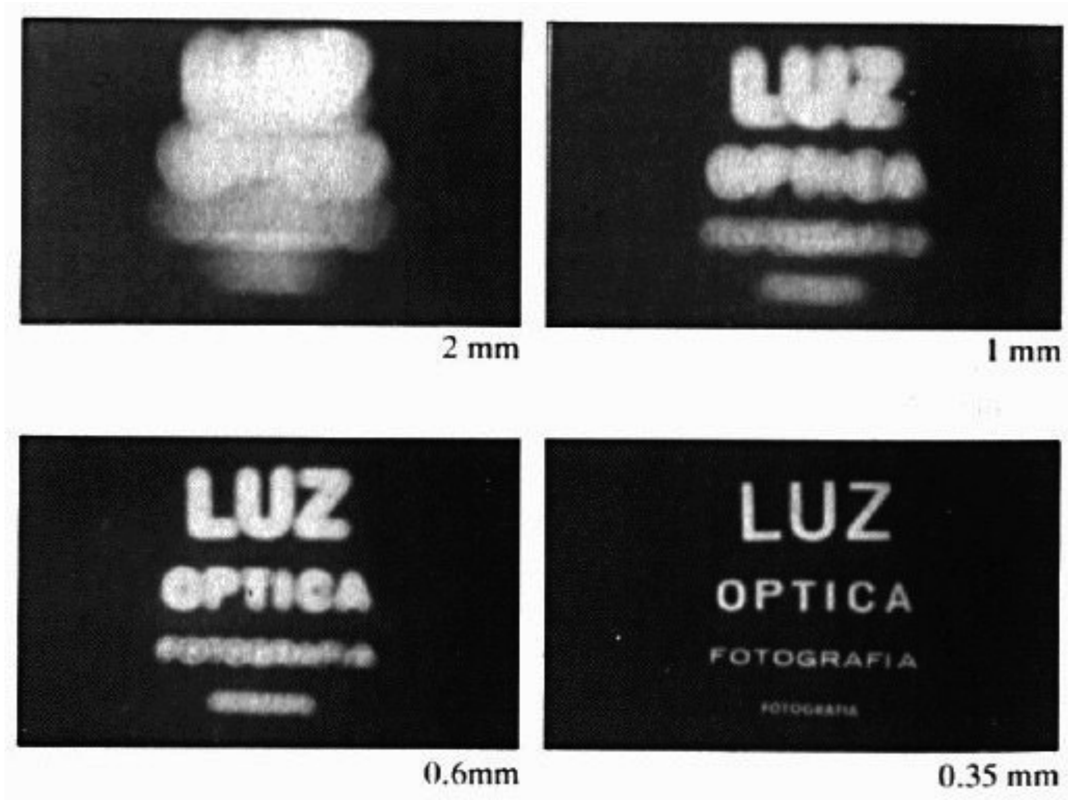
Home-made pinhole camera

Why so blurry?

Slide by A. Efros

http://www.debevec.org/Pinhole/

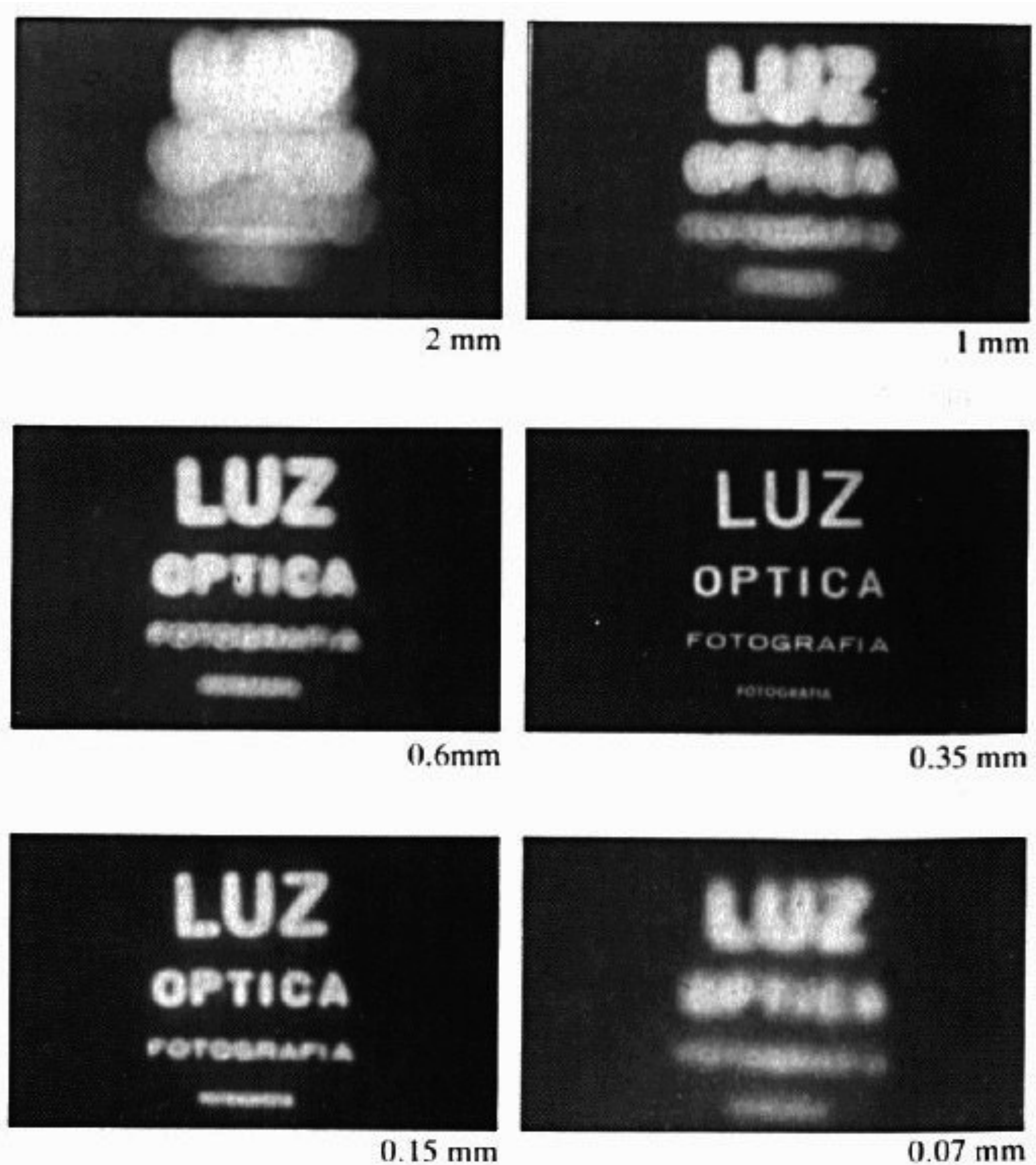
Shrinking the aperture



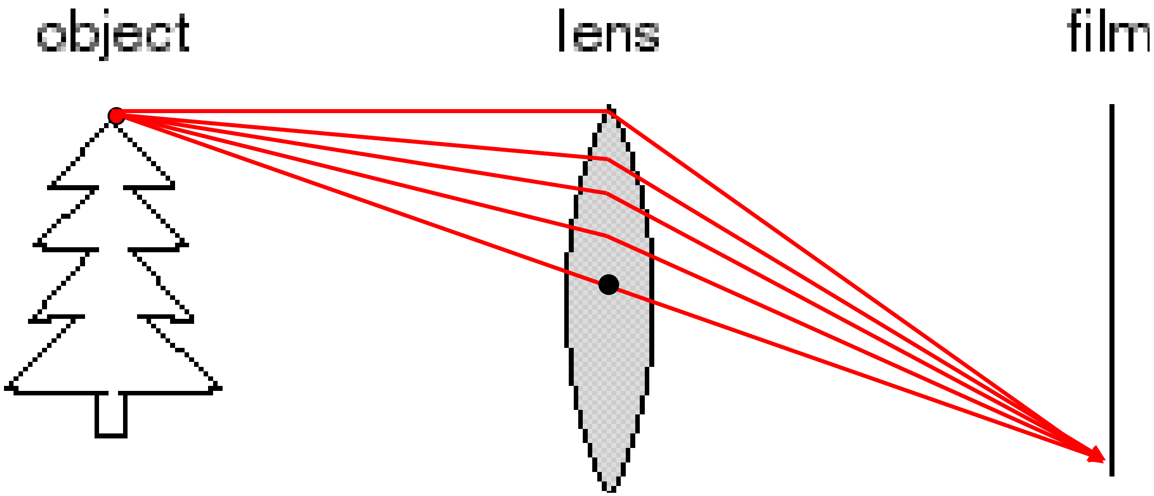
- Why not make the aperture as small as possible?
  - Less light gets through
  - Diffraction effects…

Slide by Steve Seitz

Shrinking the aperture

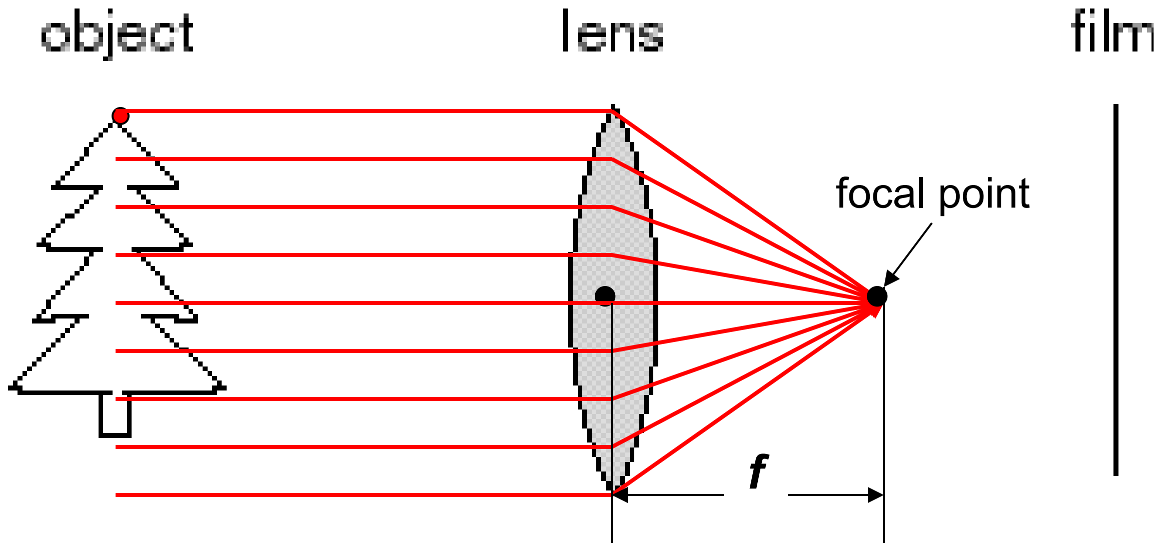


Adding a lens



- A lens focuses light onto the film
  - Rays passing through the center are not deviated

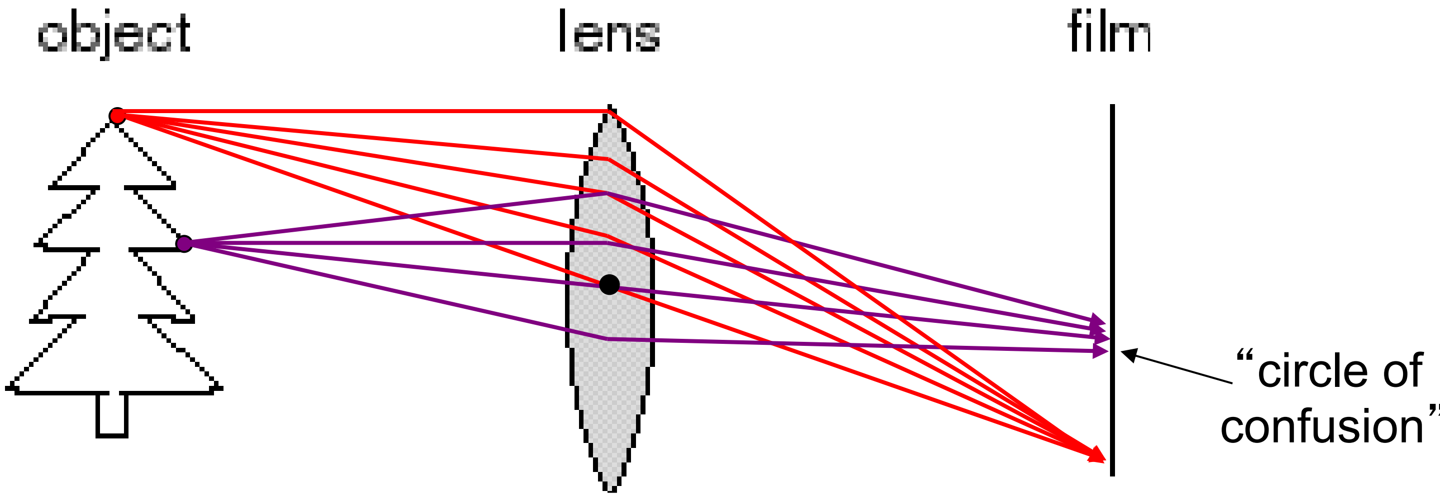
Adding a lens



- A lens focuses light onto the film
  - Rays passing through the center are not deviated
  - All parallel rays converge to one point on a plane located at the *focal length f*

Slide by Steve Seitz

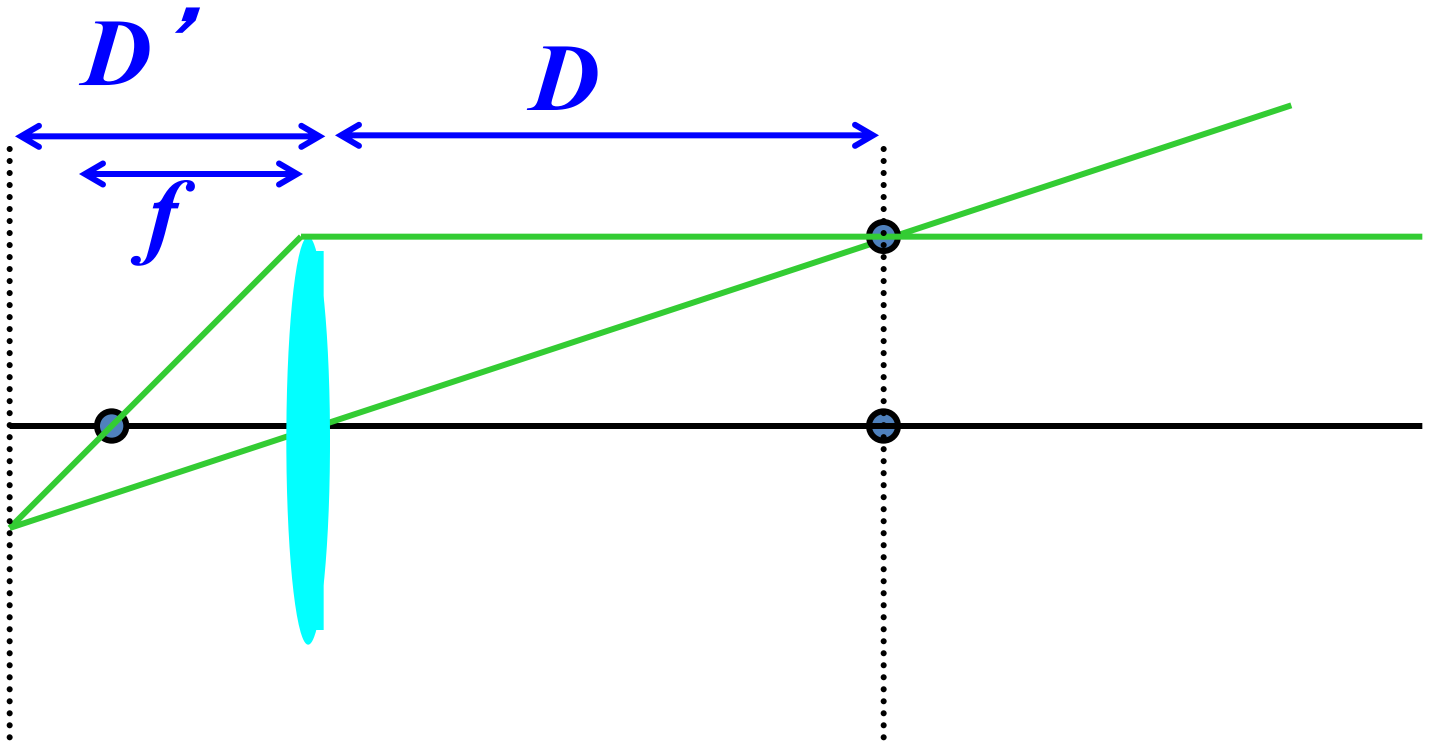
Adding a lens



- A lens focuses light onto the film
  - There is a specific distance at which objects are “in focus”
    - other points project to a “circle of confusion” in the image

Slide by Steve Seitz

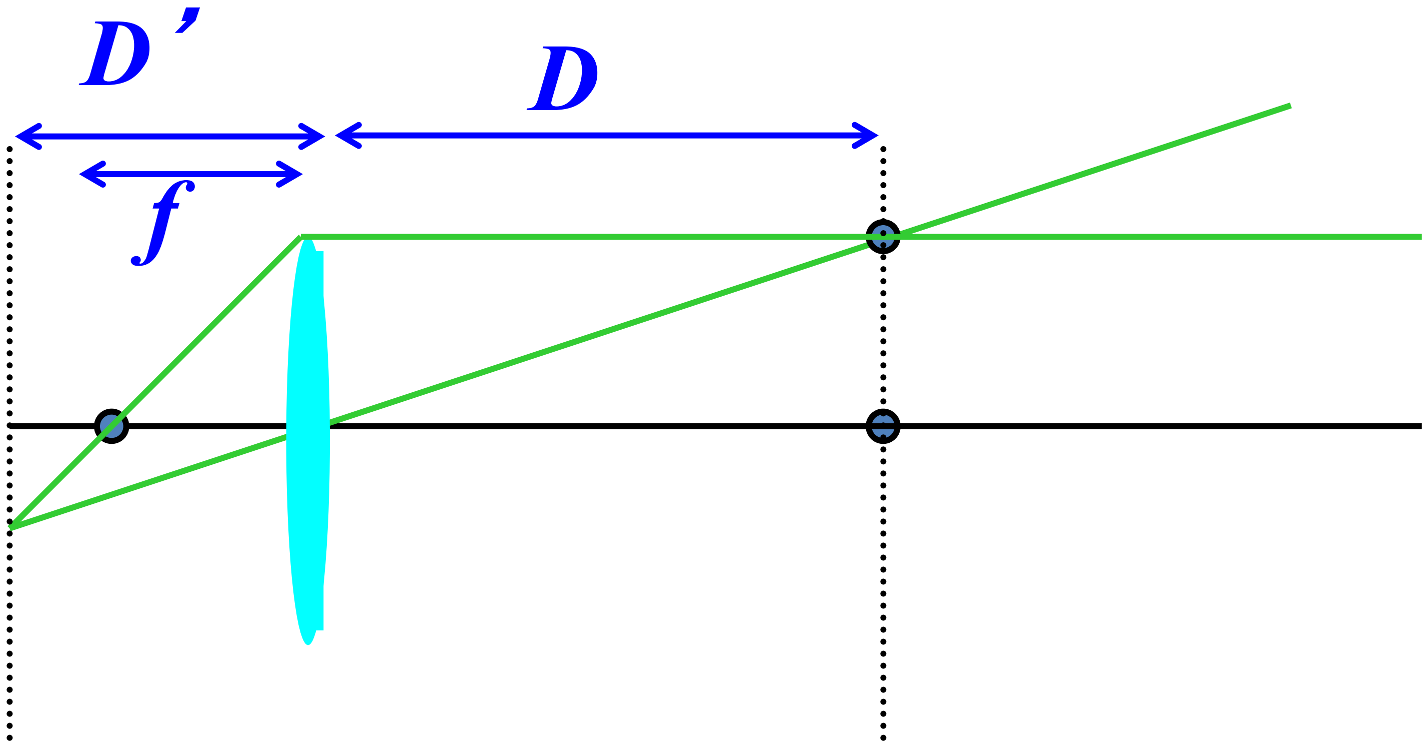
Thin lens formula



Frédo Durand's slide

Thin lens formula

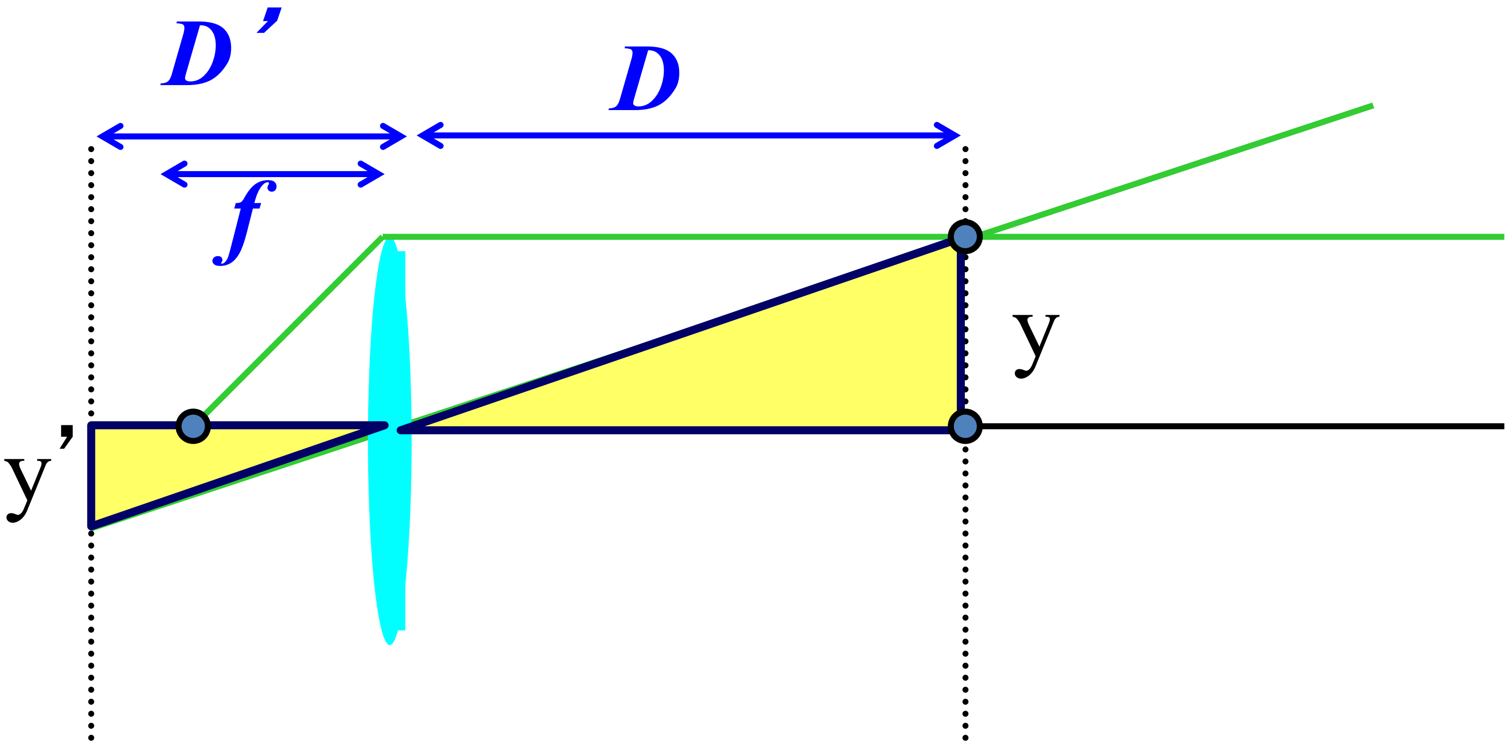
Similar triangles everywhere!



Frédo Durand's slide

Thin lens formula

$$y'/y = D'/D$$

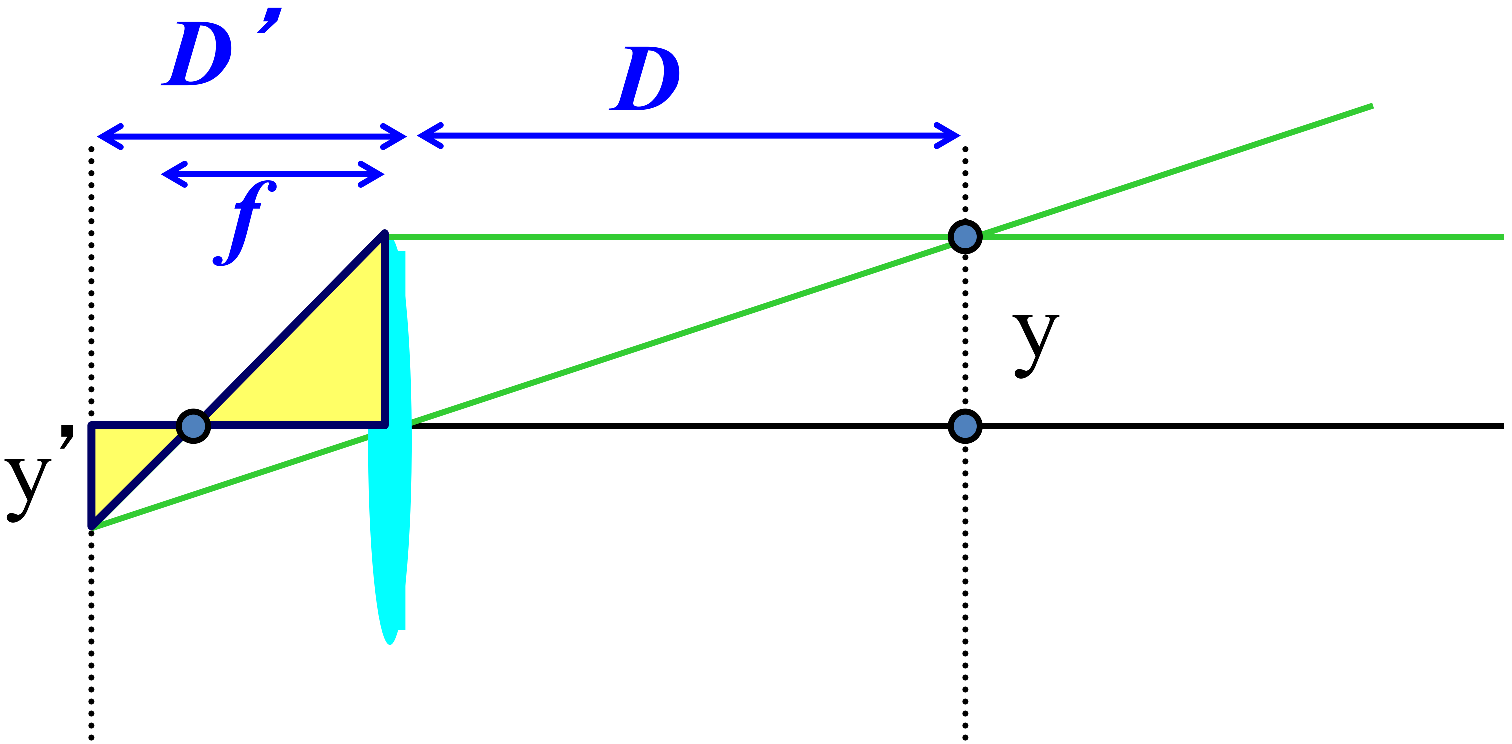


Frédo Durand's slide

Thin lens formula

$$y'/y = D'/D$$

$$y'/y = (D'-f)/f$$

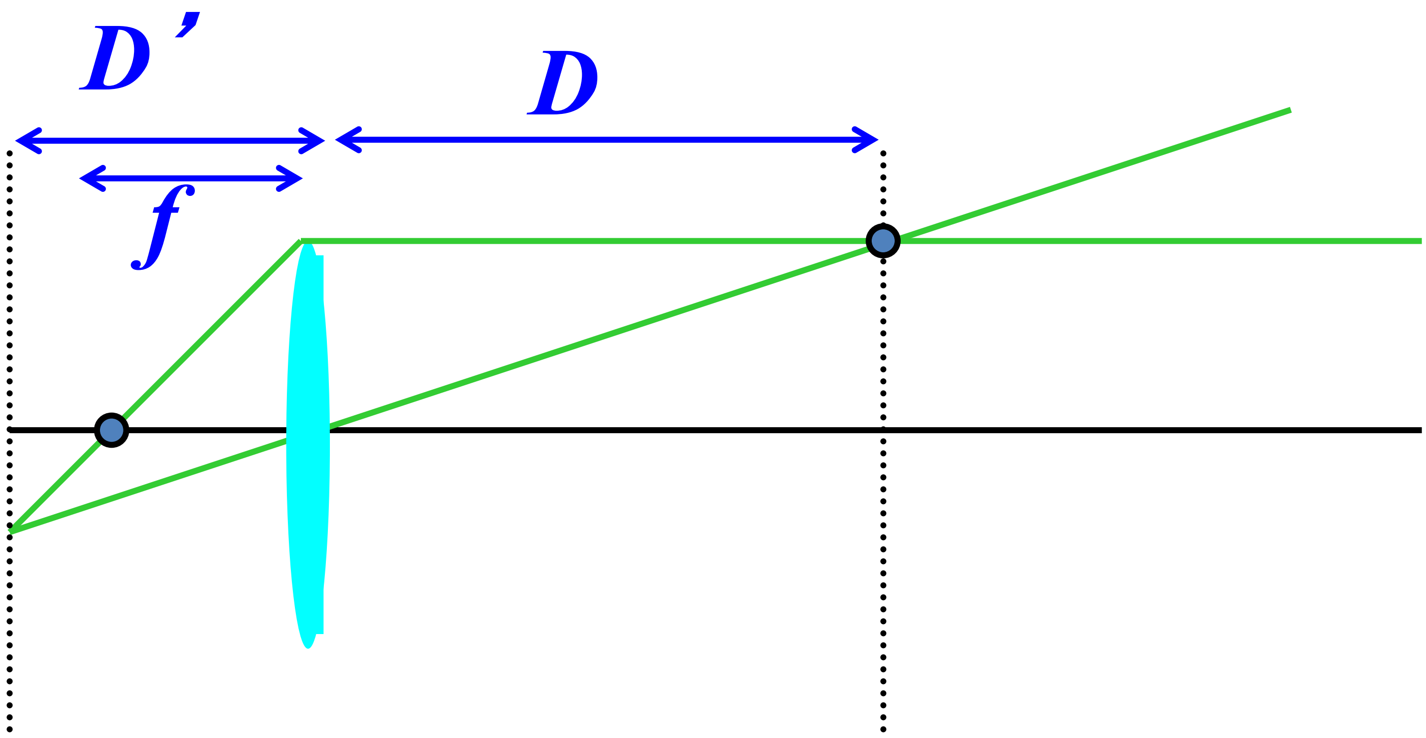


Frédo Durand's slide

Thin lens formula

$$\boxed{\frac{1}{D'} + \frac{1}{D} = \frac{1}{f}}$$

Any point satisfying the thin lens equation is in focus.



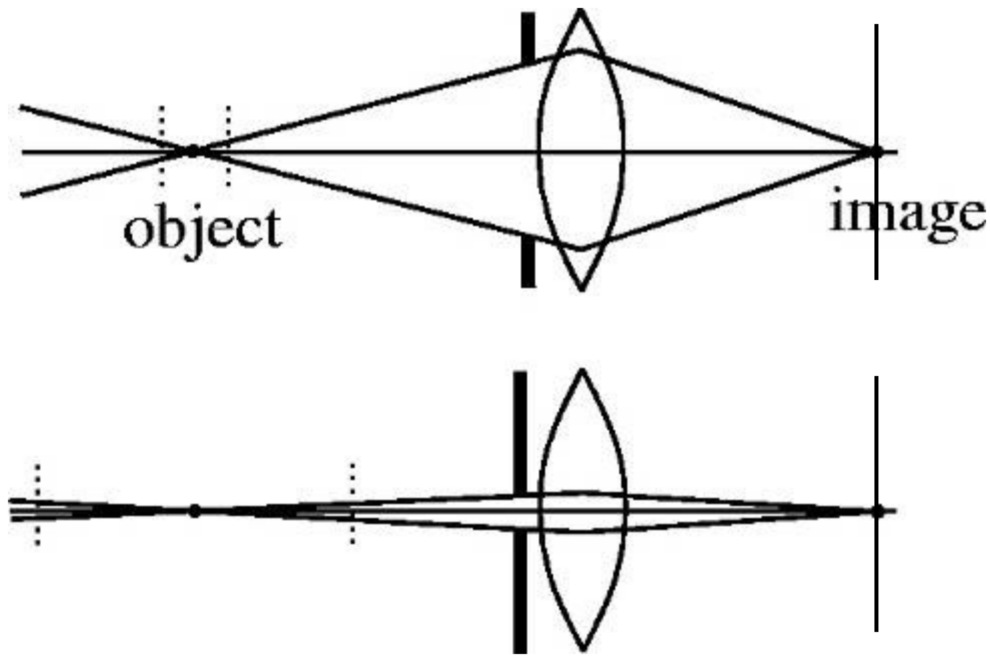
Frédo Durand's slide

Depth of Field

http://www.cambridgeincolour.com/tutorials/depth-of-field.htm

Slide by A. Efros

How can we control the depth of field?



- Changing the aperture size affects depth of field
  - A smaller aperture increases the range in which the object is approximately in focus
  - But small aperture reduces amount of light – need to increase exposure

Slide by A. Efros

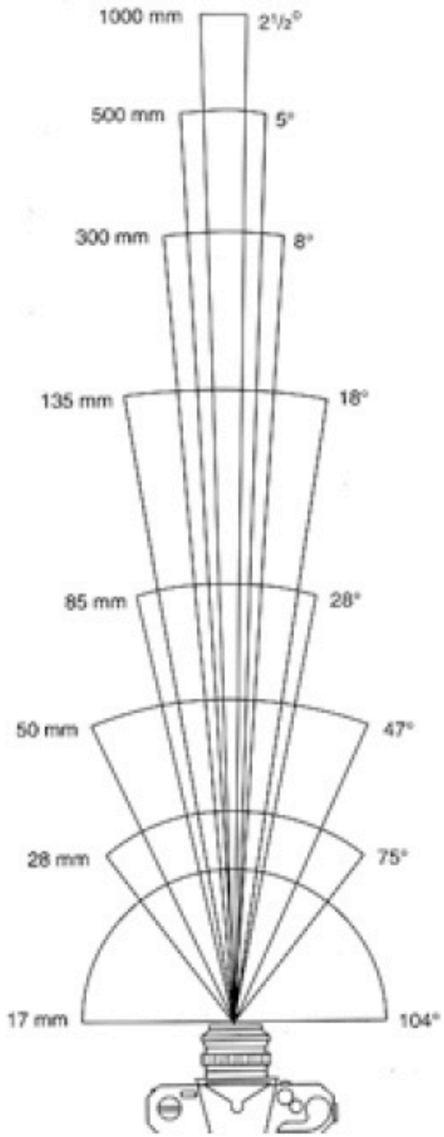
Varying the aperture

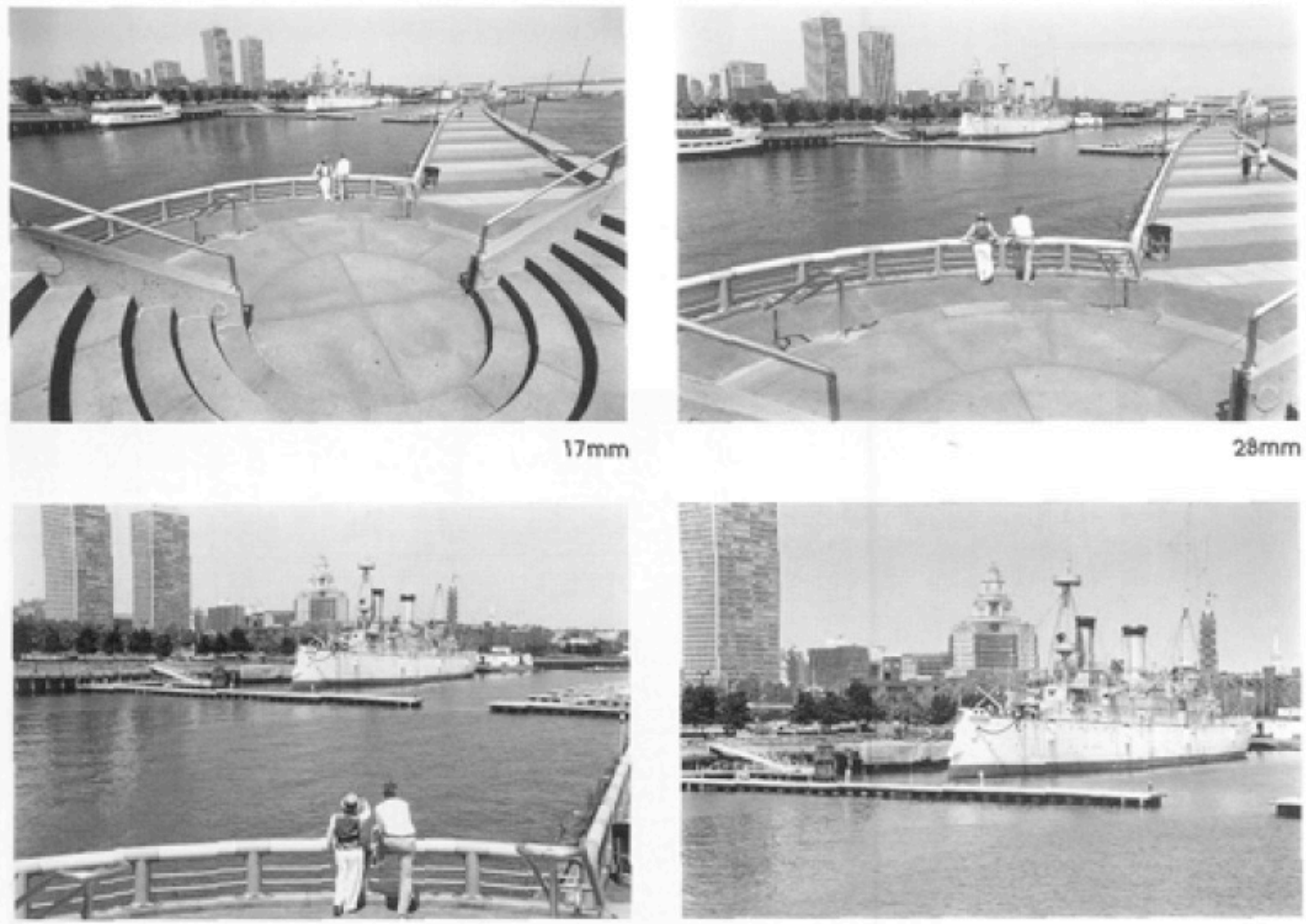
Large aperture = small DOF

Small aperture = large DOF

Slide by A. Efros

Field of View





17mm

28mm

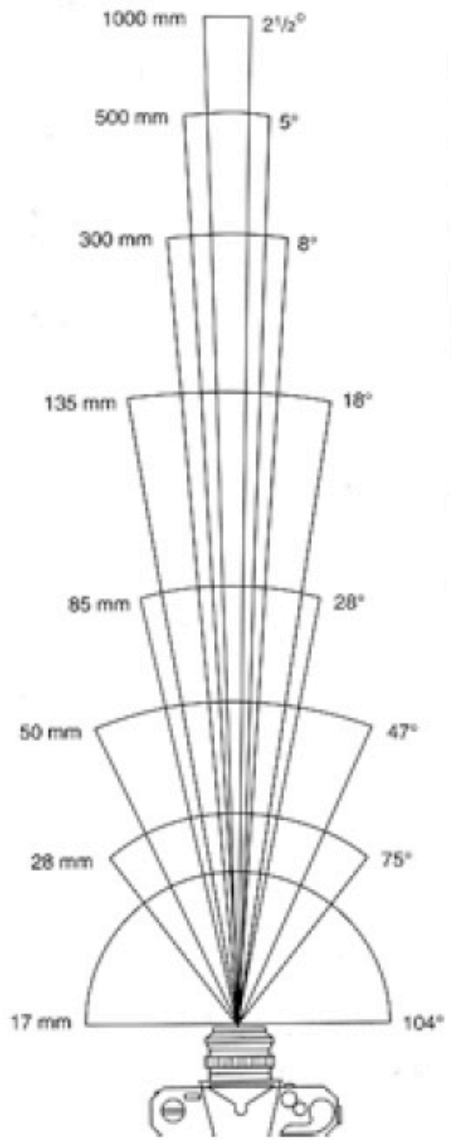
50mm

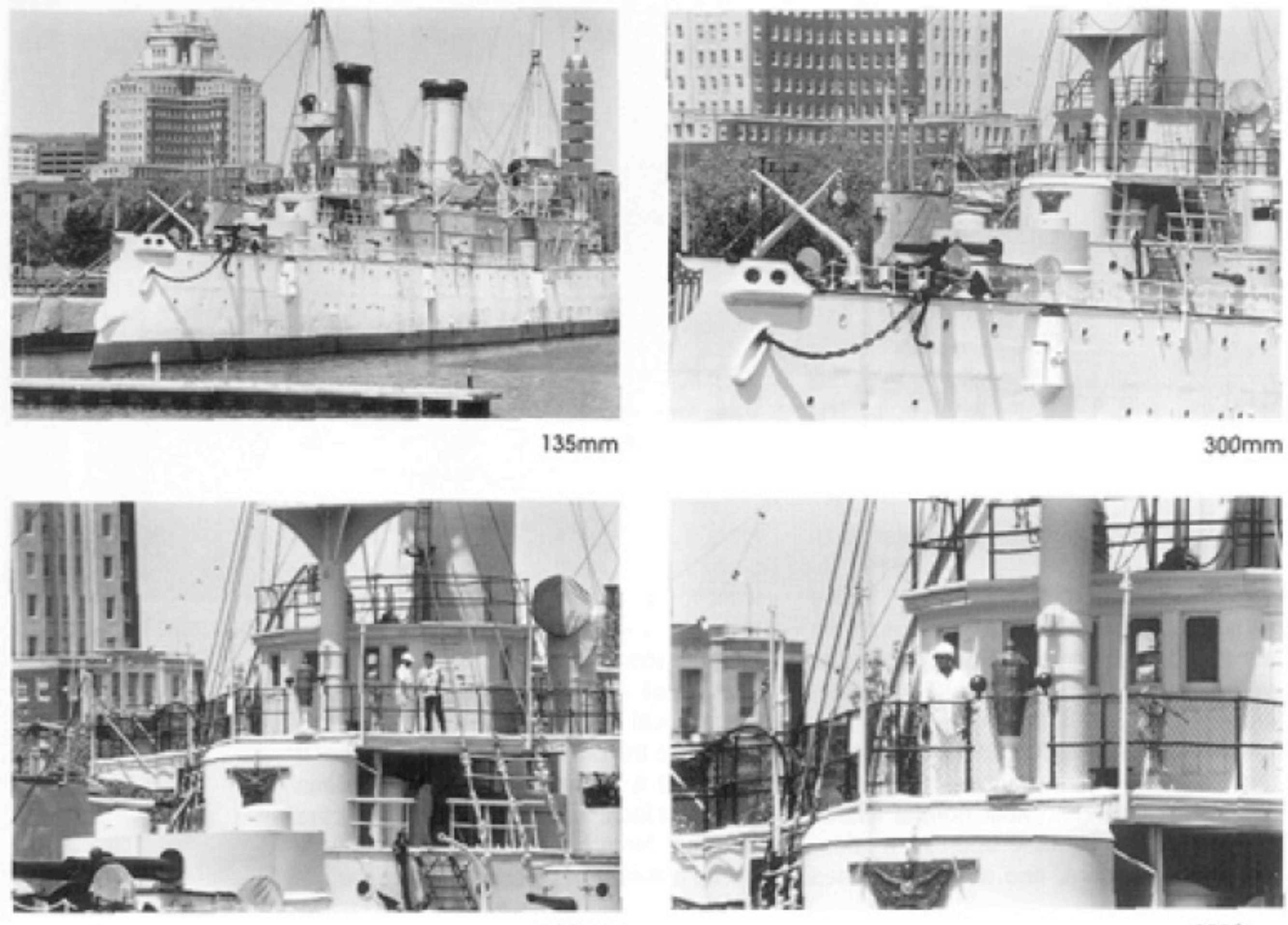
85mm

From London and Upton

Slide by A. Efros

Field of View

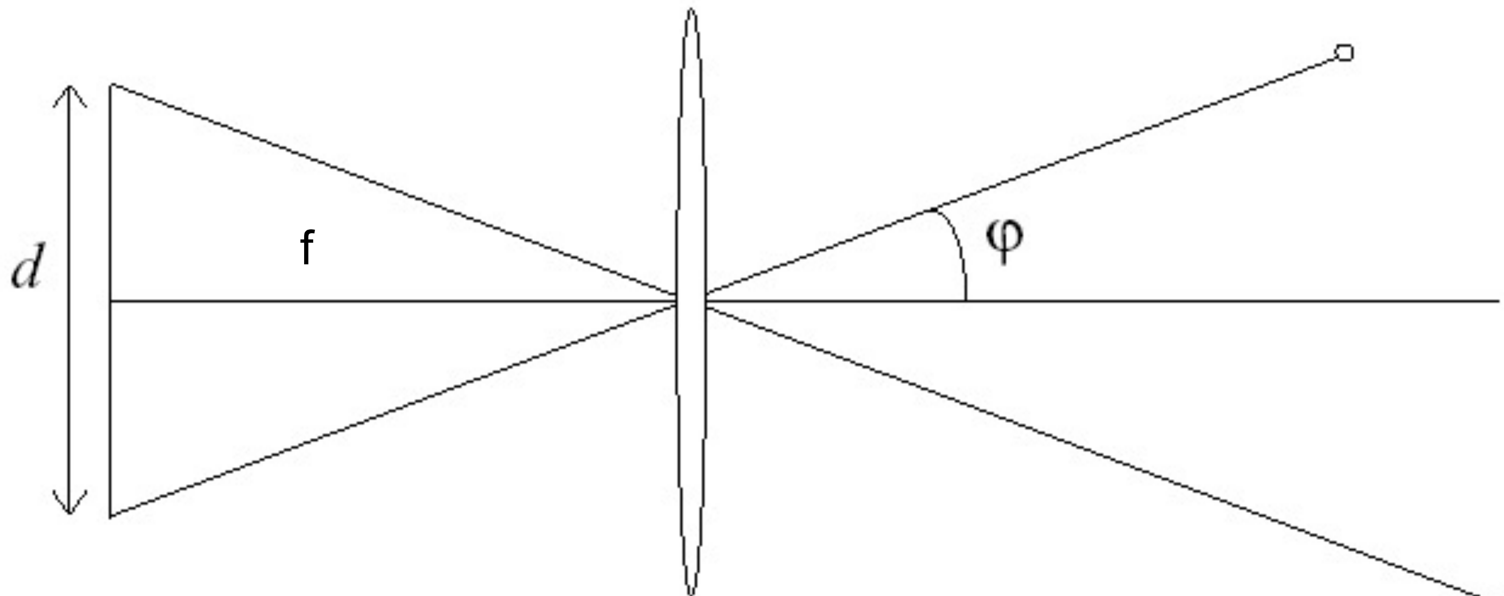




From London and Upton

Slide by A. Efros

Field of View



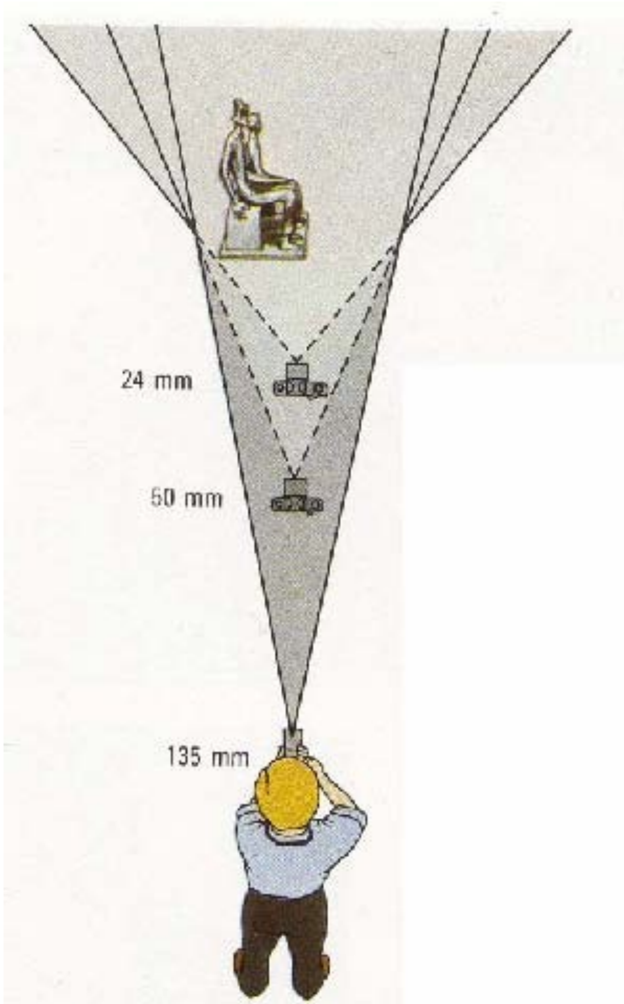
FOV depends on focal length and size of the camera retina

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

Smaller FOV = larger Focal Length

Slide by A. Efros

Field of View / Focal Length



Large FOV, small f
Camera close to car

Small FOV, large f
Camera far from the car

Sources: A. Efros, F. Durand

Same effect for faces

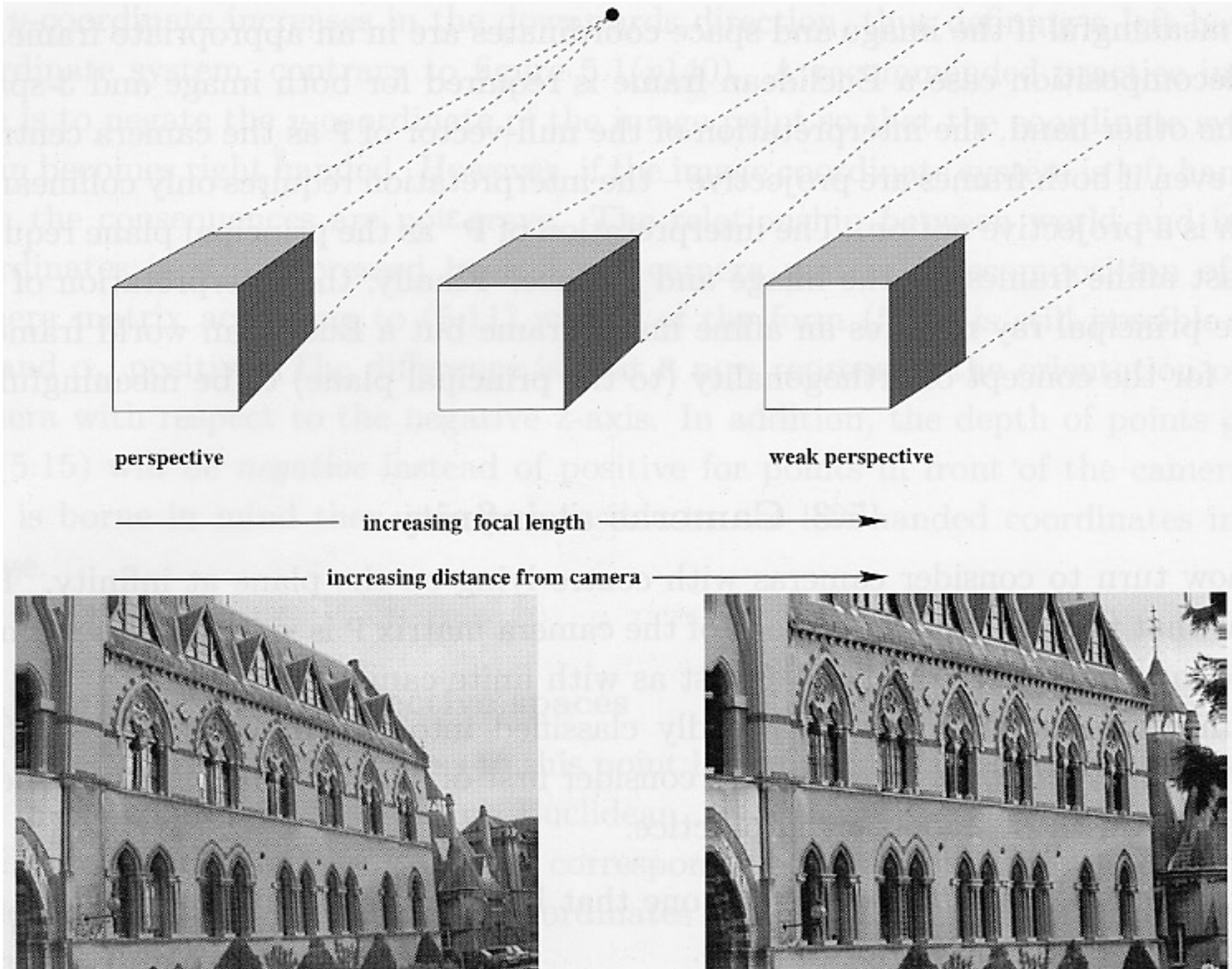
wide-angle

standard

telephoto

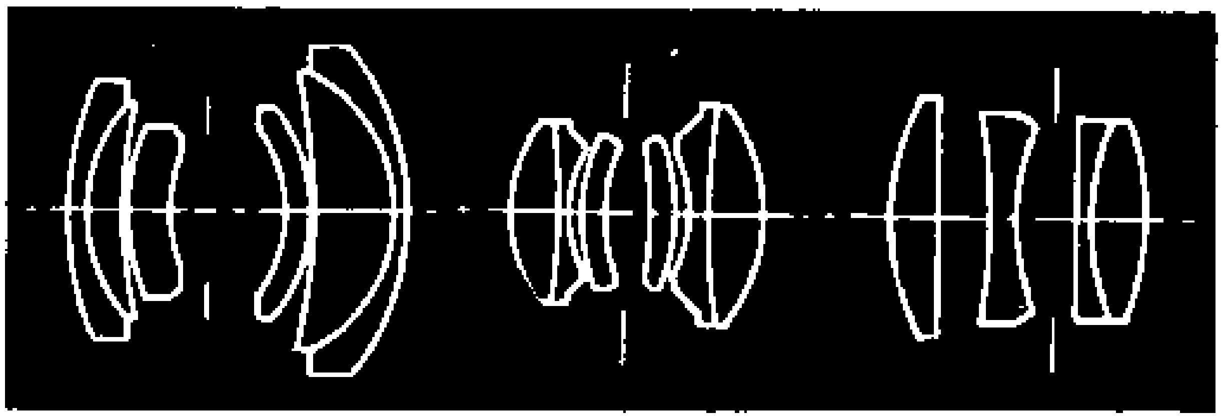
Source: F. Durand

Approximating an affine camera



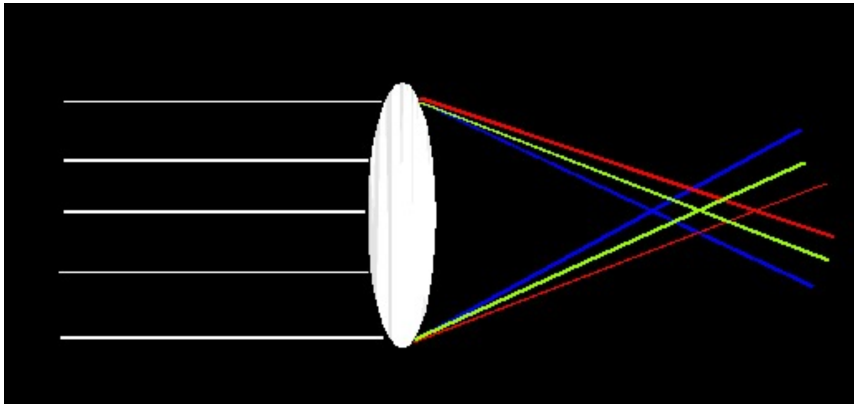
Source: Hartley & Zisserman

Real lenses



Lens Flaws: Chromatic Aberration

- Lens has different refractive indices for different wavelengths: causes color fringing

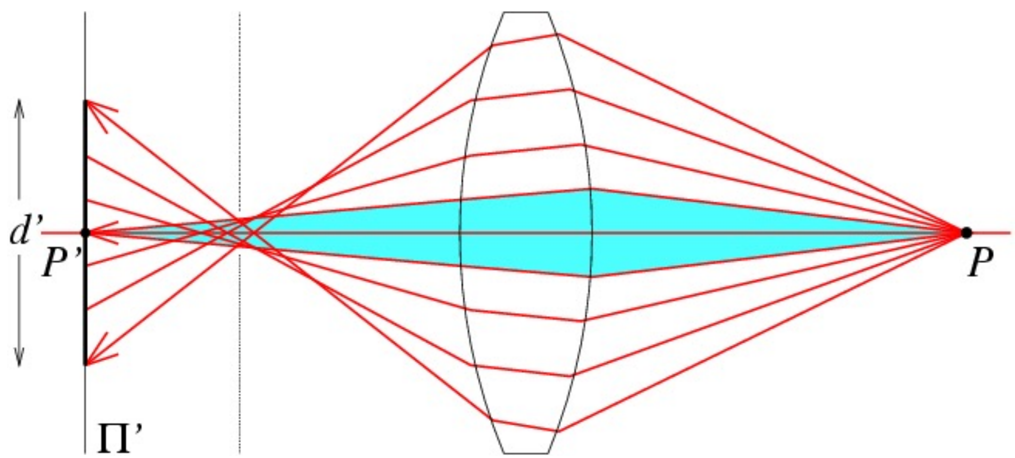


Near Lens Center

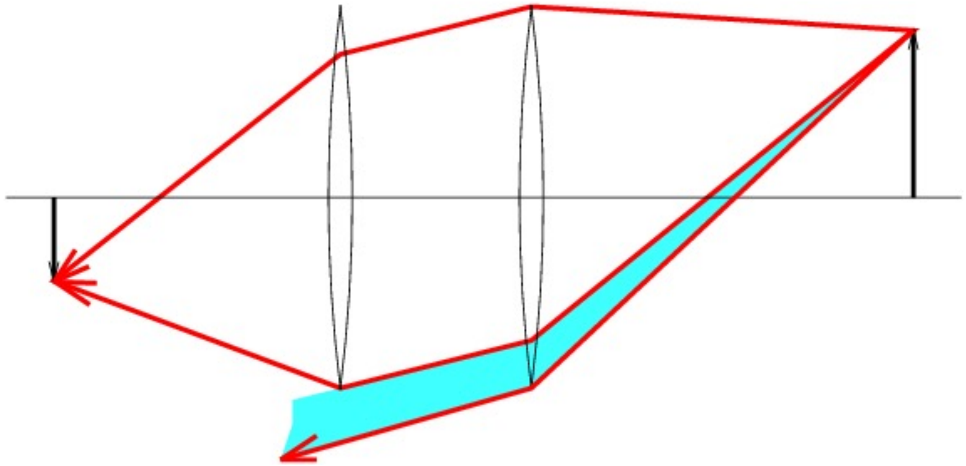
Near Lens Outer Edge

Lens flaws: Spherical aberration

- Spherical lenses don't focus light perfectly
- Rays farther from the optical axis focus closer

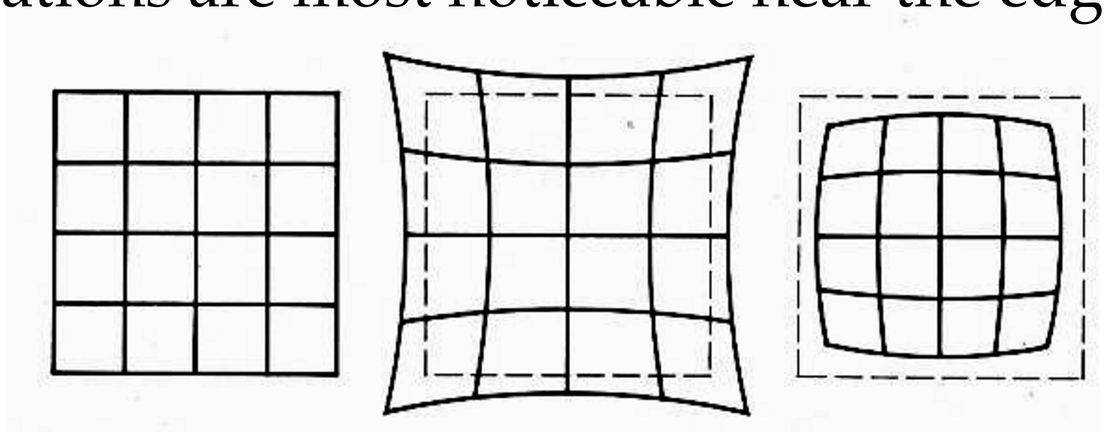


Lens flaws: Vignetting



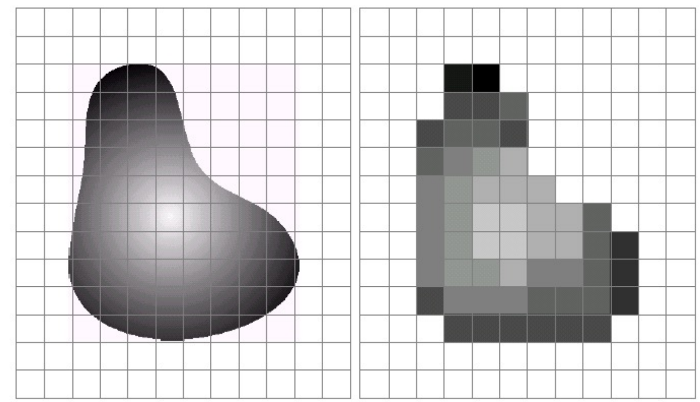
Radial Distortion

- Caused by imperfect lenses
- Deviations are most noticeable near the edge of the lens



No distortion　　Pin cushion　　Barrel

Digital camera



a b

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

- 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)
    - **Complementary metal oxide semiconductor** (CMOS)
  - http://electronics.howstuffworks.com/digital-camera.htm

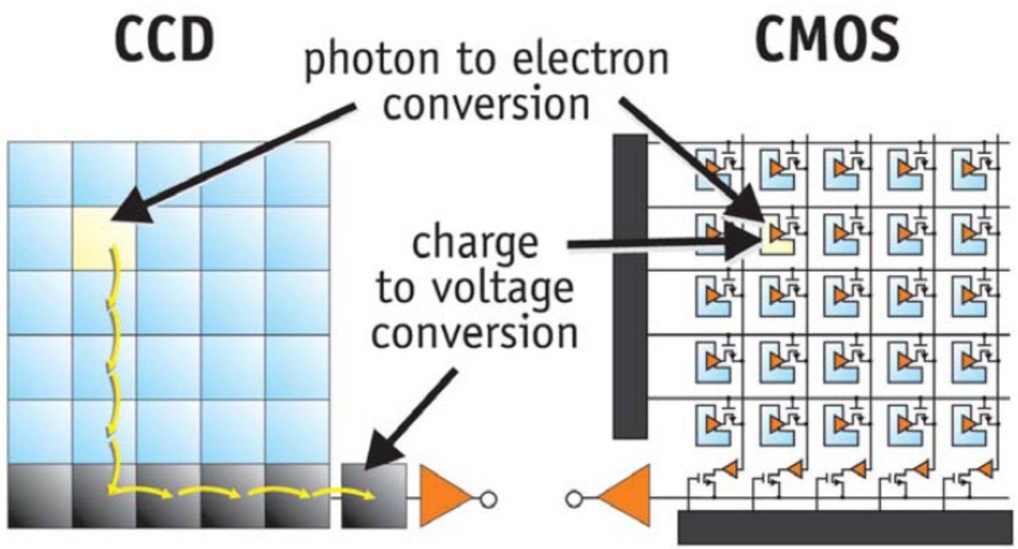
Slide by Steve Seitz

CCD vs. CMOS

- **CCD:** transports the charge across the chip and reads it at one corner of the array. An **analog-to-digital converter (ADC)** then turns each pixel's value into a digital value by measuring the amount of charge at each photosite and converting that measurement to binary form

- **CMOS:** uses several transistors at each pixel to amplify and move the charge using more traditional wires. The CMOS signal is digital, so it needs no ADC.

http://electronics.howstuffworks.com/digital-camera.htm

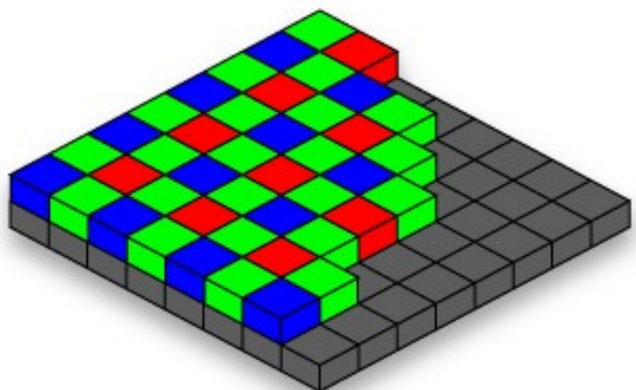


CCDs move photogenerated charge from pixel to pixel and convert it to voltage at an output node. CMOS imagers convert charge to voltage inside each pixel.

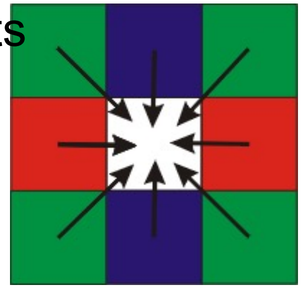
http://www.dalsa.com/shared/content/pdfs/CCD_vs_CMOS_Litwiller_2005.pdf

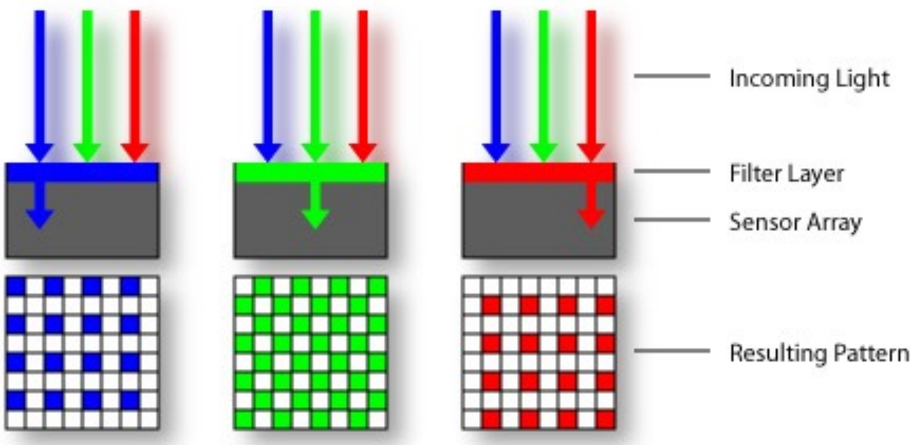
Color sensing in camera: Color filter array

Bayer grid

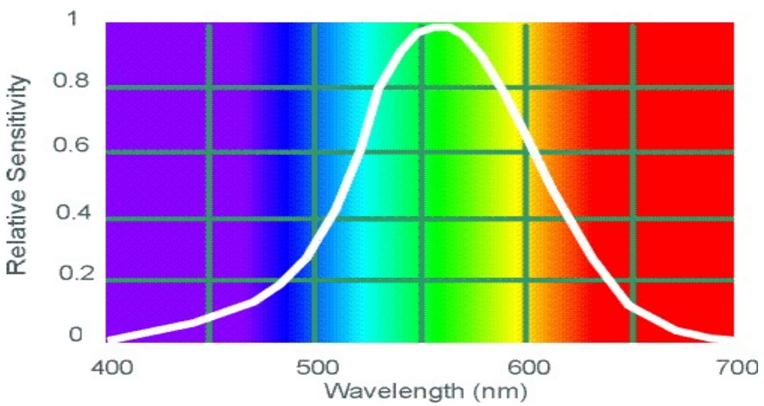


Estimate missing components from neighboring values (demosaicing)





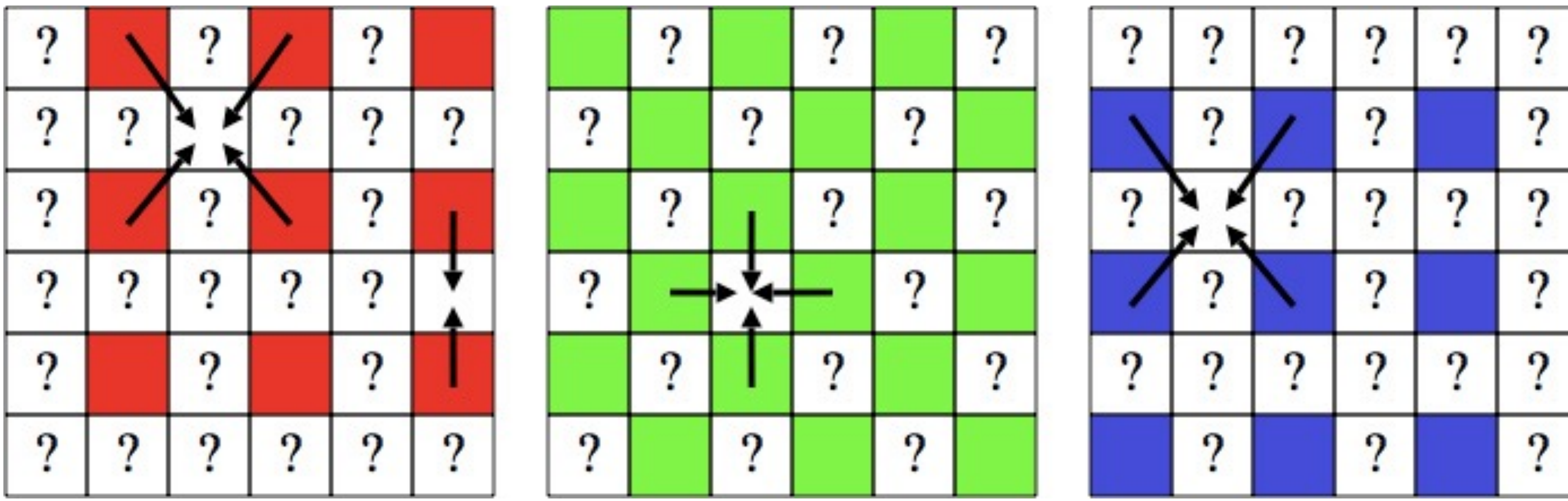
Why more green?



Human Luminance Sensitivity Function

Source: Steve Seitz

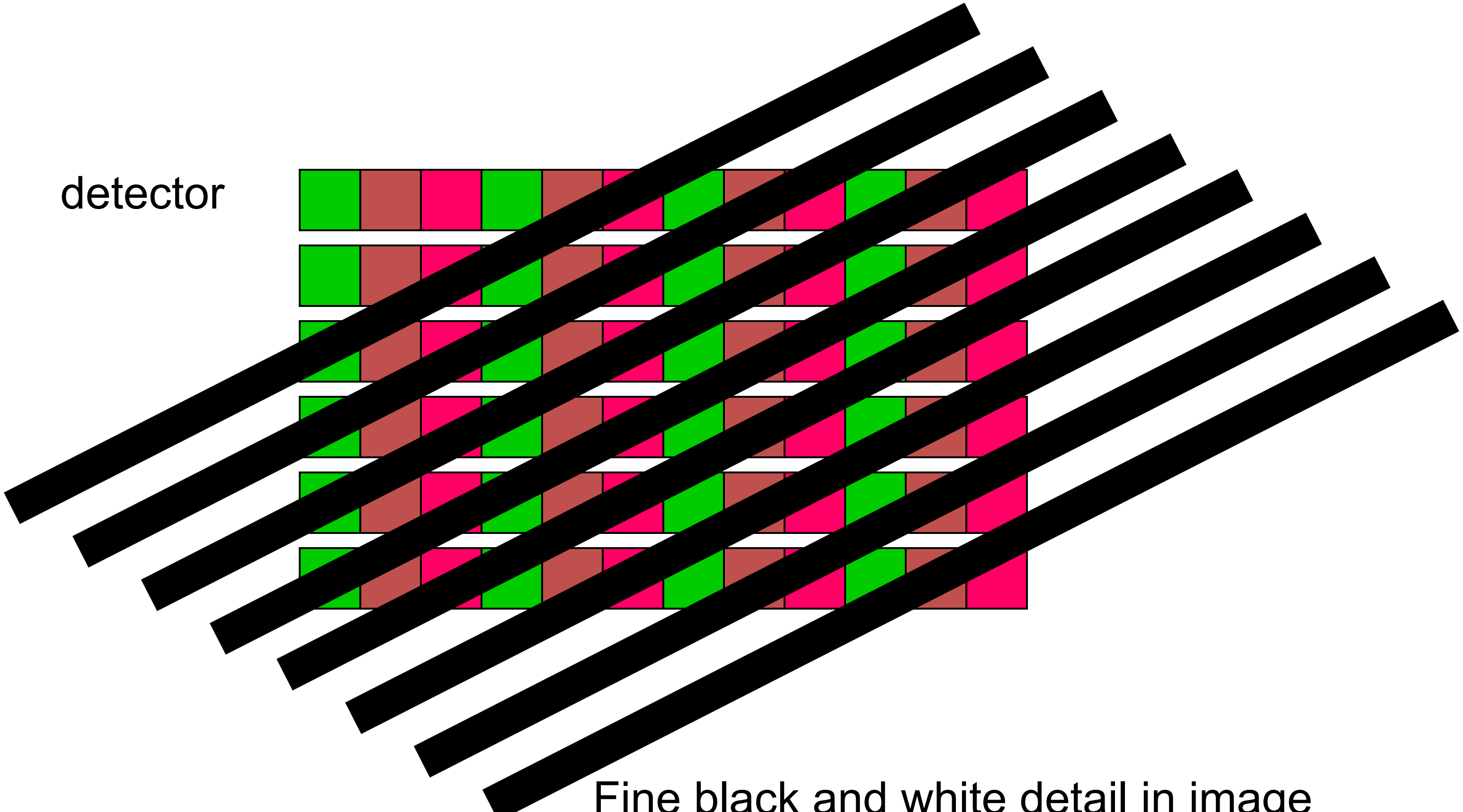
Demosaicing



Problem with demosaicing: color moire

Slide by F. Durand

The cause of color moire

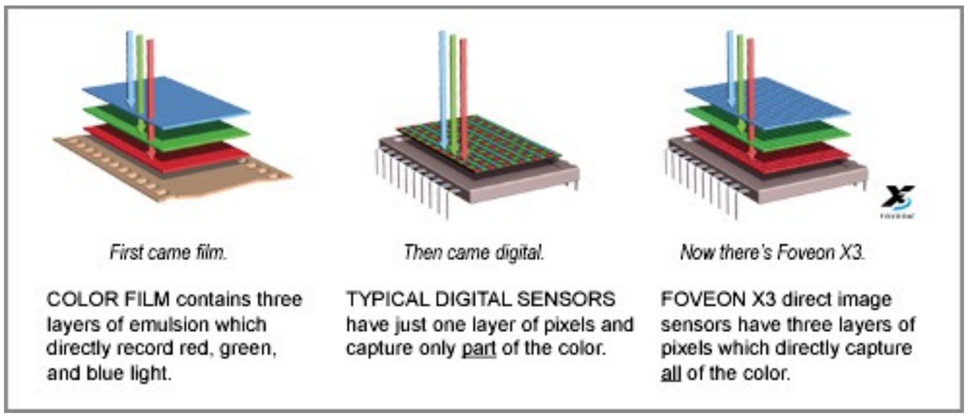


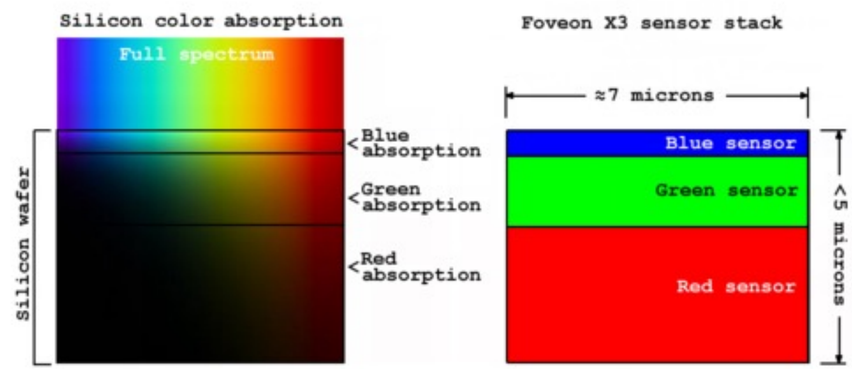
Fine black and white detail in image
misinterpreted as color information

Slide by F. Durand

Color sensing in camera: Foveon X3

- CMOS sensor
- Takes advantage of the fact that red, blue and green light penetrate silicon to different depths

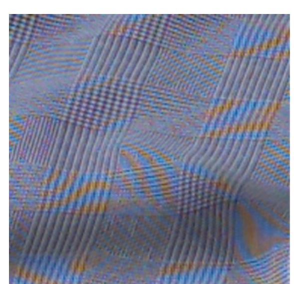
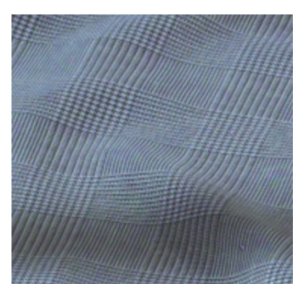




http://www.foveon.com/article.php?a=67

http://en.wikipedia.org/wiki/Foveon_X3_sensor

better image quality

Source: M. Pollefeys

Digital camera artifacts

- Noise
  - low light is where you most notice noise
  - light sensitivity (ISO) / noise tradeoff
  - stuck pixels

- In-camera processing
  - oversharpening can produce halos

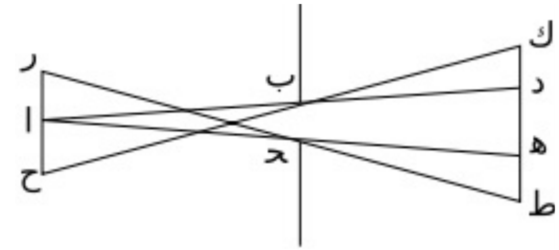
- Compression
  - JPEG artifacts, blocking
- Blooming
  - charge overflowing into neighboring pixels

- Color artifacts
  - purple fringing from microlenses,
  - white balance

Slide by Steve Seitz

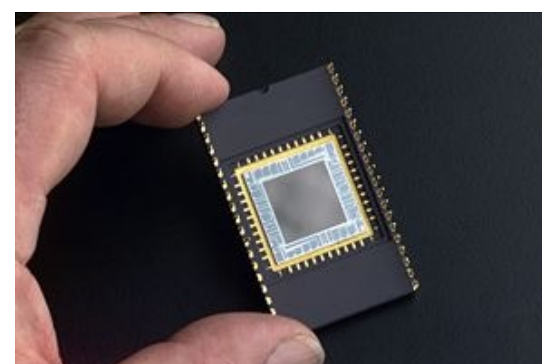
Historic milestones

- **Pinhole model:** Mozi (470-390 BCE), Aristotle (384-322 BCE)
- **Principles of optics (including lenses):** Alhacen (965-1039 CE)
- **Camera obscura:** Leonardo da Vinci (1452-1519), Johann Zahn (1631-1707)
- **First photo:** Joseph Nicephore Niepce (1822)
- **Daguerréotypes** (1839)
- **Photographic film** (Eastman, 1889)
- **Cinema** (Lumière Brothers, 1895)
- **Color Photography** (Lumière Brothers, 1908)
- **Television** (Baird, Farnsworth, Zworykin, 1920s)
- **First consumer camera with CCD**: Sony Mavica (1981)
- **First fully digital camera:** Kodak DCS100 (1990)



Alhacen’s notes

Niepce, “La Table Servie,” 1822



CCD chip