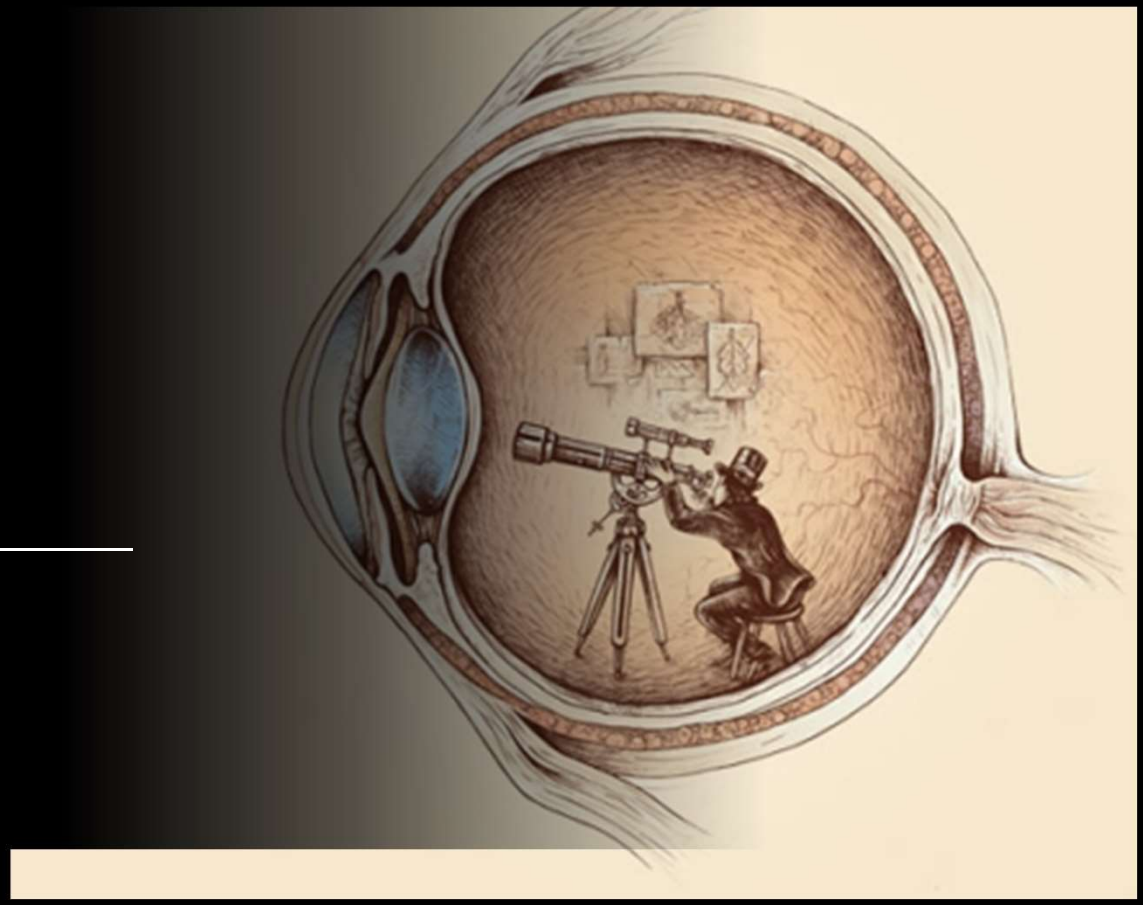
CS216:
Image
Understanding
(**Recognition**)

Lecture 1

**Alex Berg**



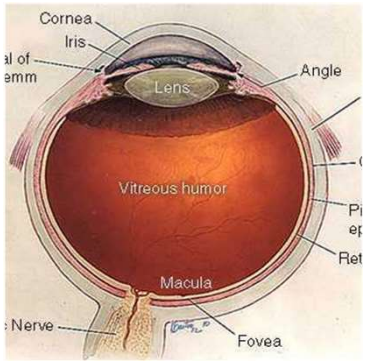
# Today quick intro to the types of things in the class

- Some examples of recognition in computer vision
- Syllabus and course structure
- Some discussion and thought question on image formation
- Read 1-3 (optionally Chapter 7) of Szeliski Computer Vision book for next class, but
- We will talk today about some things related to filtering, e.g. natural image statistics
- Hopefully the reading will clear things up if needed.

# Why Vision?

# Why Vision? Light!

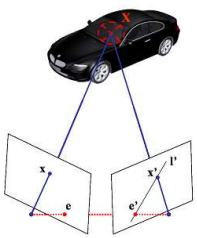
It is how we see other people, navigate our environment, communicate ideas, entertain, and **measure** the world around us.




# Why is light good for measurement?

Microscopy

Surveillance



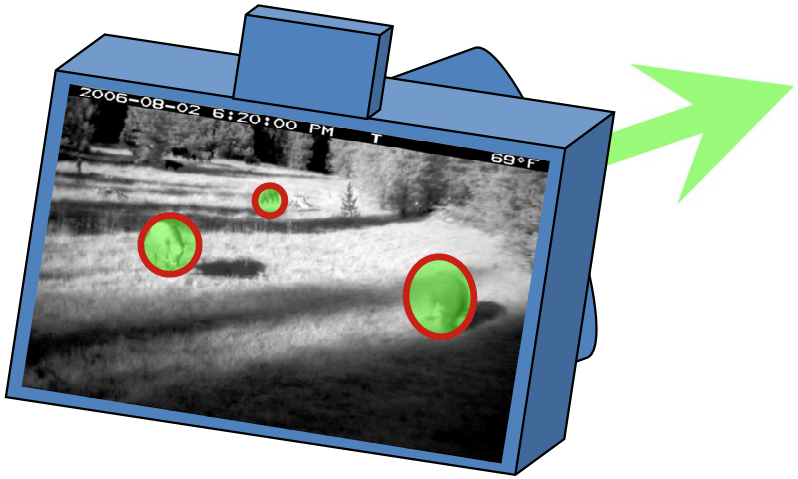
3D Analysis / Navigation

Remote Sensing

- Plentiful, sometimes free
- Interacts with many things, but not too many
- Goes generally straight over distance
- Very small → high spatial resolution
- Fast, but not too fast → time of flight sensors
- Easy to detect → cameras work, are cheap
- Comes in flavors ( wavelengths )

# Why Computational Visual Recognition?

Because we need to know which bits to measure!




McIlroy, Allen-Diaz, Berg
Journal of Rangeland Ecology and Management 2011

3 Cows at 6:20 PM

Need to recognize which parts of the image are cows, deer, humans, grass, shadows, etc.

# What is there to recognize?

http://www.flickr.com/photos/sgcallawayimages/3306849049/

Increasing structural complexity

Dog

Single label

Dog, Tree, Fence, Leaves, Autumn

Multiple labels


Localization/ Parsing

**“Happy shaggy Airedale poses in the autumn forest.”**

Description

# Novel view synthesis for scenes as a recognition problem!

Recognition of known shape to guess approximate geometry

Map the input image onto geometry and render from new view

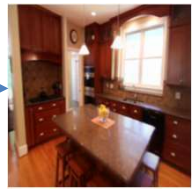
(a) input image

(c) novel views

Real Image 1

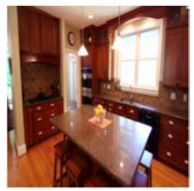
Model

Estimated depth



Rendered From Image 2 camera

Difference



Real Image 2

**World Sheet Revisited**
Ronghang Hu, Nikhila Ravi, Alex Berg, Deepak Pathak
2021

# Course structure

Current Syllabus (will change in response to progress/interest) will be posted on Canvas

Reference text: Szeliski second edition available free online https://szeliski.org/Book/

1. Intro and image formation, start on image statistics
   **Read/review Chapter 1-3 from Szeliski book (optionally look at chapter 7)**
2. Low-level vision: continuation of prev & edges, histogram features, correspondence, optical flow
3. Classification and beginning of statistical machine learning
4. Statistical machine learning and deep learning (homework 2)
5. Progression on structured prediction: Detection, Pose Estimation
6. Progression on structure prediction: Segmentation (homework 3)
7. Vision and language and datasets (written test)
8. Synthesis NeRF, Dalle-2, Diffusion models (homework 4)
9. Embodied vision / Video
10. Looking back at the dreams of the past for the future of computer vision

# Grading structure

30% Homeworks 1-3
30% Test
20% Homework 4
20% Participation (quizzes, class, office hour, other interaction)

General expectation of ~10 hours a week per course, so lectures + ~8 hours.

Participate in class, do reading, do very well on the assignments and test -> A+
**Particpate in class, do reading, do all the assignments and test reasonably -> A**
Come to class, do some reading, do some of all the assignments and test -> B
Academic integrity or honor code violation -> Bad grade

**Do your own work**, discussion is encouraged, but do all your own work/writing. No copying and cite sources, collaborators, and discussions.

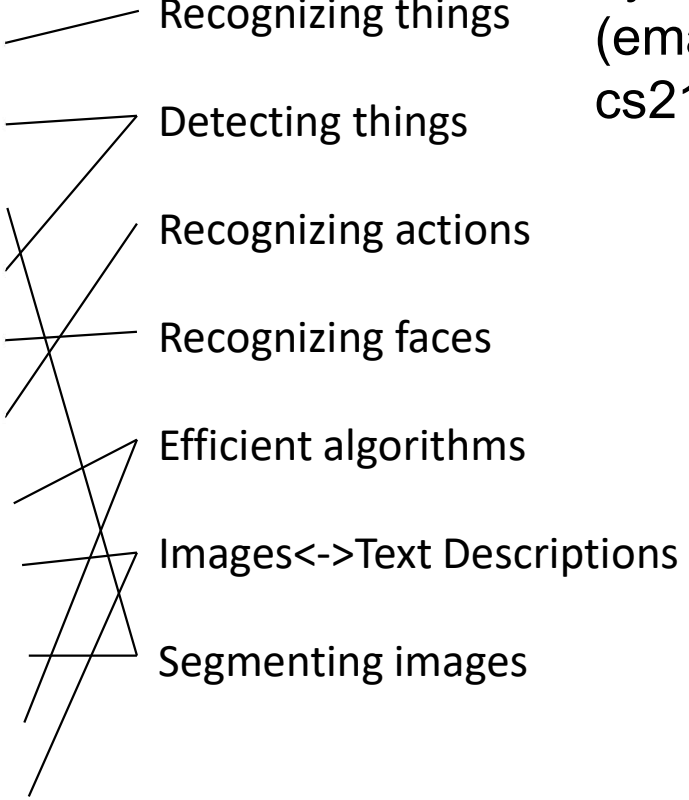
# Professor structure

Alexander C Berg

FOLLOW

Professor of Computer Science, University of California Irvine
Verified email at uci.edu - Homepage
computer vision machine learning visual perception web mining

| TITLE | CITED BY | YEAR |
|---|---|---|
| Imagenet large scale visual recognition challenge<br>O Russakovsky, J Deng, H Su, J Krause, S Satheesh, S Ma, Z Huang, ...<br>International journal of computer vision 115, 211-252 | 48825 | 2015 |
| Ssd: Single shot multibox detector<br>W Liu, D Anguelov, D Erhan, C Szegedy, S Reed, CY Fu, AC Berg<br>Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The ... | 45111 | 2016 |
| Segment anything<br>A Kirillov, E Mintun, N Ravi, H Mao, C Rolland, L Gustafson, T Xiao, ...<br>Proceedings of the IEEE/CVF International Conference on Computer Vision ... | 7783 | 2023 |
| Dssd: Deconvolutional single shot detector<br>CY Fu, W Liu, A Ranga, A Tyagi, AC Berg<br>arXiv preprint arXiv:1701.06659 | 2792 | 2017 |
| Attribute and simile classifiers for face verification<br>N Kumar, AC Berg, PN Belhumeur, SK Nayar<br>2009 IEEE 12th international conference on computer vision, 365-372 | 2033 | 2009 |
| Recognizing action at a distance<br>Efros, Berg, Mori, Malik<br>Proceedings Ninth IEEE International Conference on Computer Vision, 726-733 ... | 1862 | 2003 |
| SVM-KNN: Discriminative nearest neighbor classification for visual category recognition<br>H Zhang, AC Berg, M Maire, J Malik<br>2006 IEEE Computer Society Conference on Computer Vision and Pattern ... | 1725 | 2006 |
| Babytalk: Understanding and generating simple image descriptions<br>G Kulkarni, V Premraj, V Ordonez, S Dhar, S Li, Y Choi, AC Berg, TL Berg<br>IEEE transactions on pattern analysis and machine intelligence 35 (12), 2891 ... | 1686 | 2013 |
| Parsenet: Looking wider to see better<br>W Liu<br>arXiv preprint arXiv:1506.04579 | 1545 | 2015 |
| Classification using intersection kernel support vector machines is efficient<br>S Maji, AC Berg, J Malik<br>2008 IEEE conference on computer vision and pattern recognition, 1-8 | 1380 | 2008 |
| Modeling context in referring expressions<br>L Yu, P Poirson, S Yang, AC Berg, TL Berg<br>Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The ... | 1271 | 2016 |




bergac@uci.edu

Office hours
THUR 1-2 in
DBH 4204 and
by appointment
(email me w/
cs216 in subject)

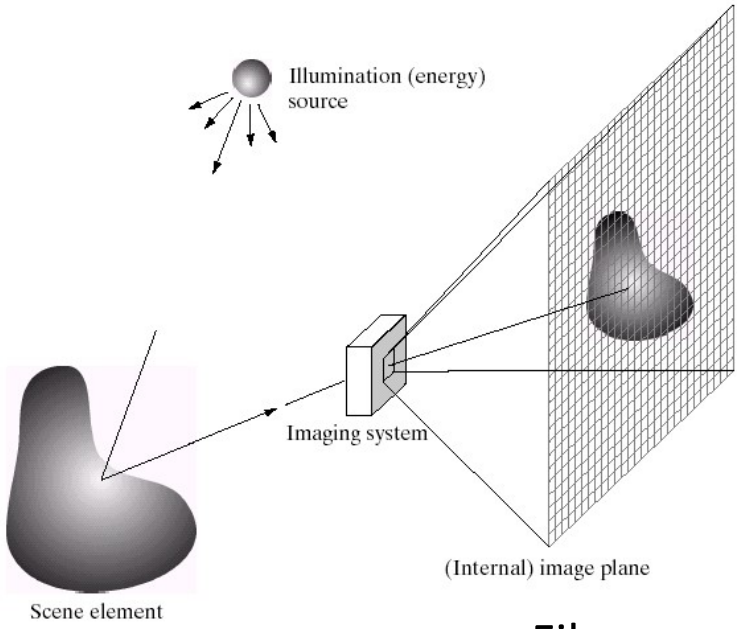
# Student structure

Take a few minutes

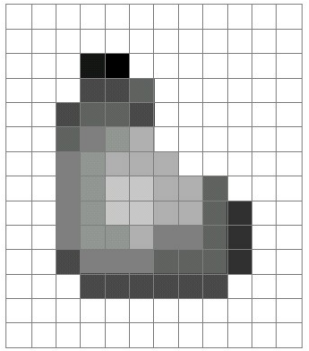
Department:
Research Area:
Year:
1-2 things you want from the course:
Any other comments:

Write on paper (anonymous is fine)
or email to bergac@uci.edu

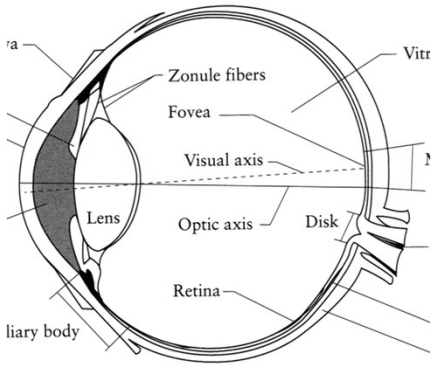
# Image Formation




Film

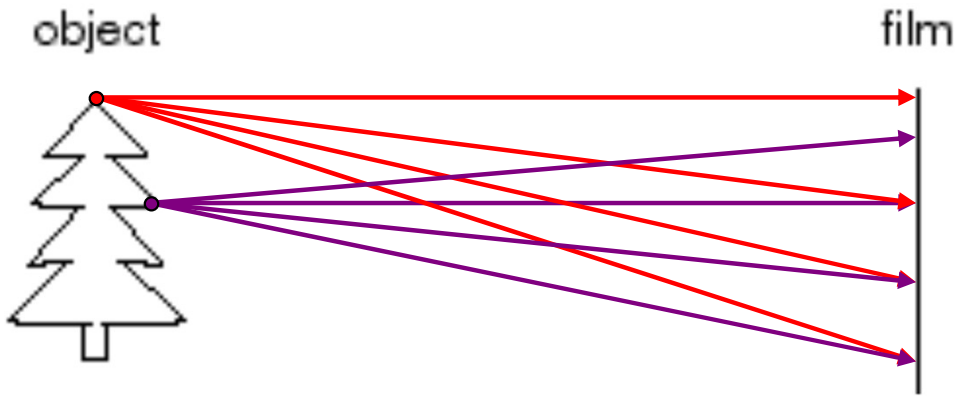


Digital Camera




The Eye

# How do we see the world?

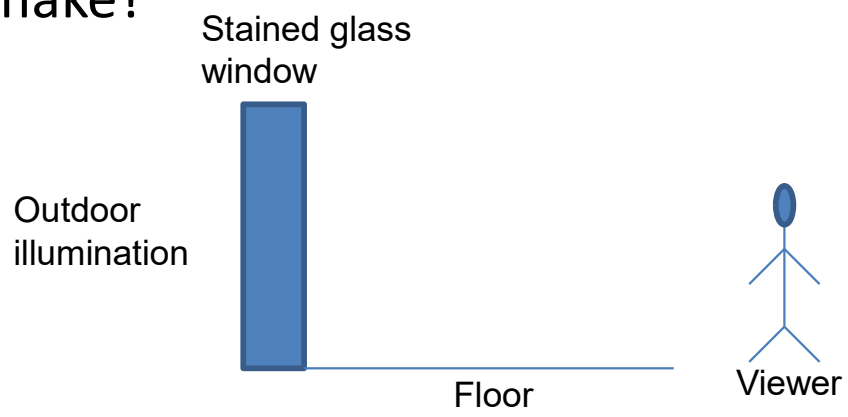



- If we put a piece of film in front of an object do we get an “image” ?

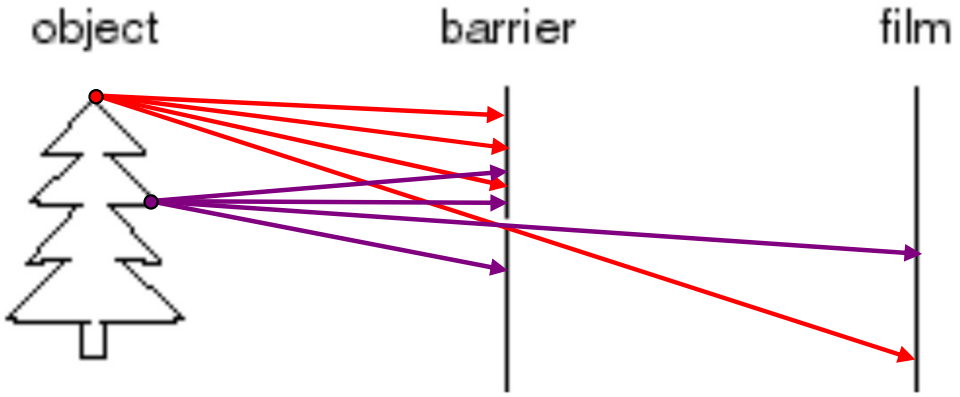
Slide by Steve Seitz

# Thought experiment (reading study questions)

- How would you write code to render what you would see on the ground in front of a stain glass window?
  - How to model light coming through the window?
  - How to integrate light that hits a point on the ground?
  - How to model what that looks like to a viewer?
  - What simplifying assumptions do you make?
  - What would the image look like?



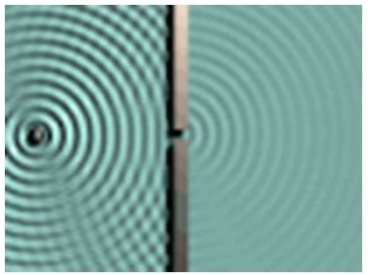
# Pinhole camera



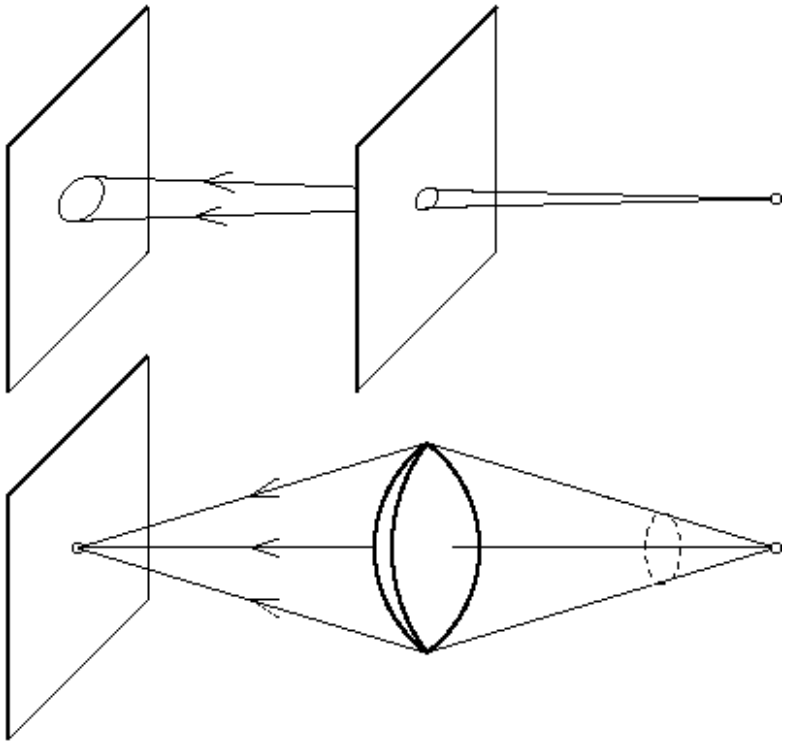

- Add a barrier to block off most of the rays
  - Each point on the film only "sees" light from a single point in the world
  - Keeps everything from being blurred together
  - The "pinhole" opening known as the **aperture**

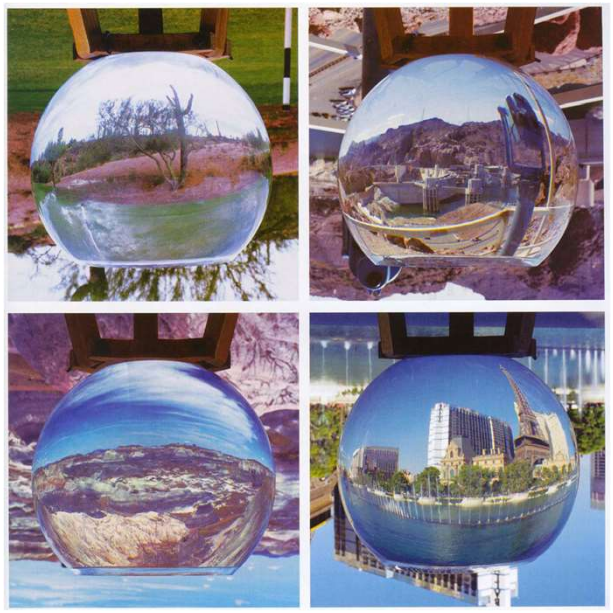
Slide by Steve Seitz

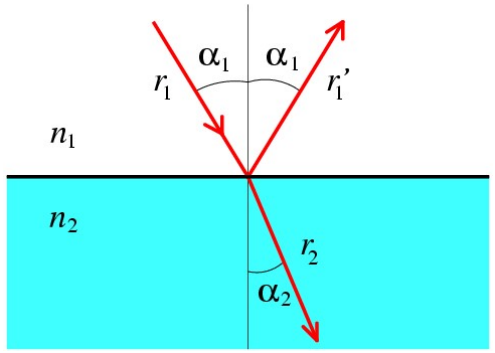
# Shrinking the pinhole

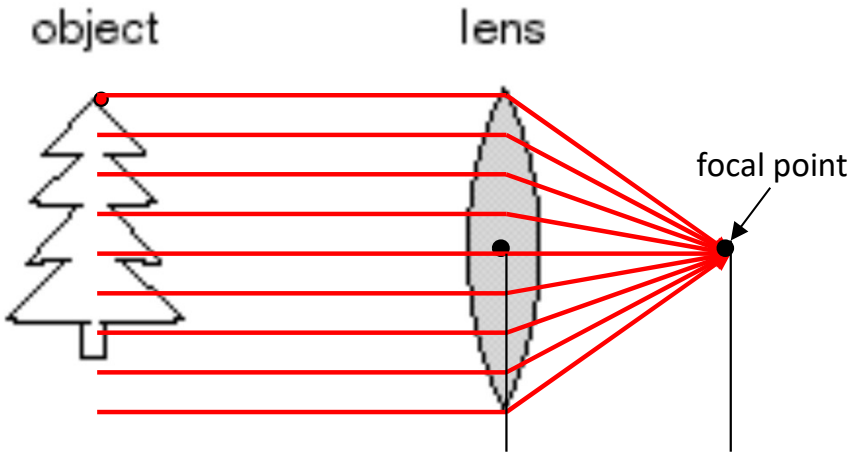
# Practical solution: Glass lenses that refract light





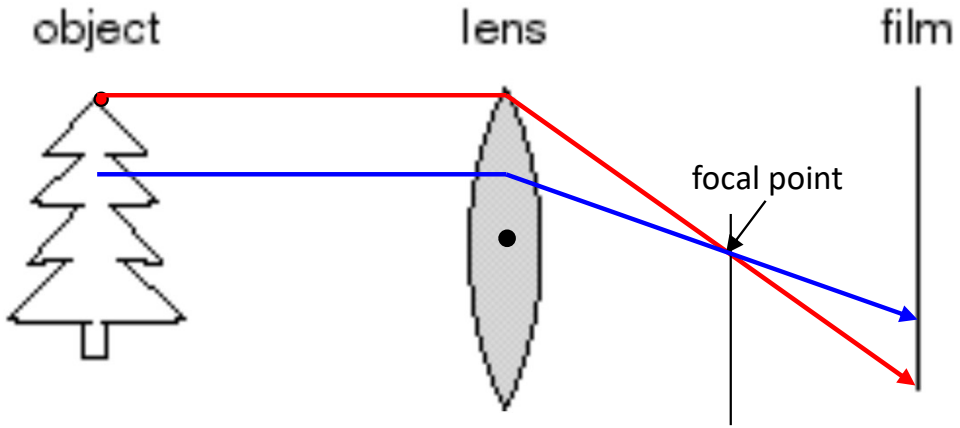


# Adding a lens




- Ideal lens focuses light convergently
  - All parallel rays converge to one point on a plane located at the *focal point* of the lens

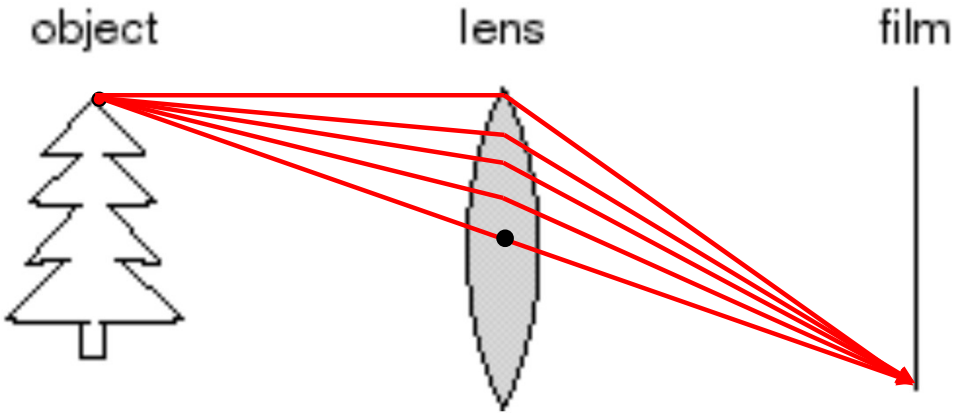
# Adding a lens




Ideal lens focuses light convergently

- All parallel rays converge to one point on a plane located at the *focal point* of the lens

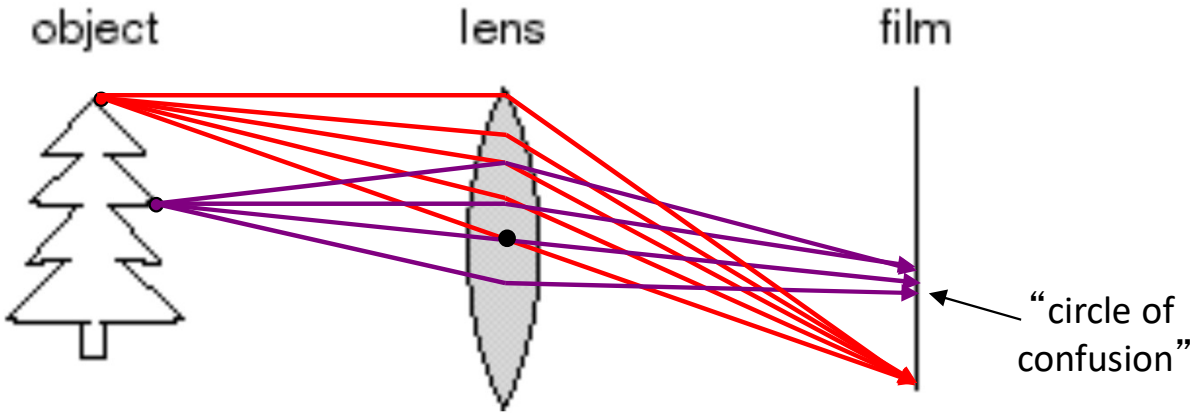
# Adding a lens




A lens focuses light onto the film

- All parallel rays converge to one point on a plane located at the *focal point* of the lens
- Rays passing through the center are not deviated

# Adding a lens




- A lens focuses light onto the film
  - There is a specific distance at which objects are “in focus”
  - Other points are blurred out over the film

Slide by Steve Seitz

# Depth of Field



Slide by A. Efros

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

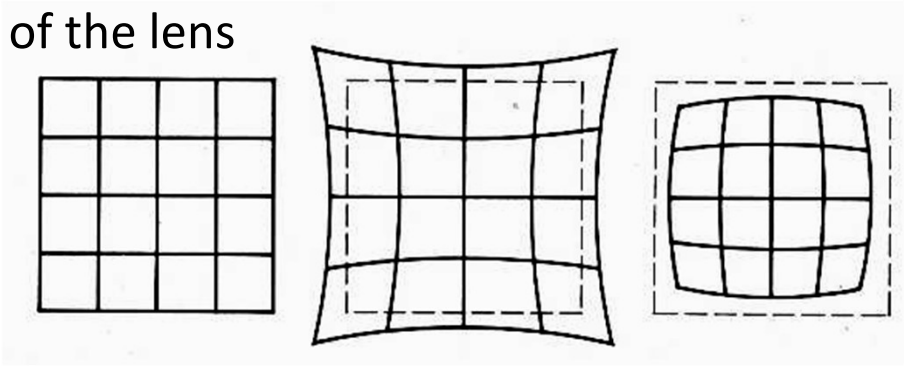
# Additional effect of aperture

- Depth of field


From Photography, London et al.

# Radial Distortion

- Caused by imperfect lenses
- Deviations are most noticeable for rays that pass through the edge of the lens



No distortion Pin cushion Barrel

Source: Steve Seitz

# Recap

- Pinhole is the simplest model of perspective image formation
- Lenses gather more light
  - But get only one plane focused
  - Focus by moving sensor/film
  - Cannot focus infinitely close
- Real lens behavior reasonably captured by perspective projection, possibly with some correction for distortion
- Questions?
- **Think about the stained glass "thought question"**
- **Read/review Szeliski book chapters 1-3 and optionally look over chapter 7.**

# Recap

- Pinhole is the simplest model of perspective image formation
- Lenses gather more light
  - ~~But get only one plane focused~~ (Not even that, perfectly)
  - Focus by moving sensor/film
  - Cannot focus infinitely close
- Real lens behavior reasonably captured by perspective projection, possibly with some correction for distortion
- Questions?
- **Think about the stained glass "thought question"**
- **Read/review Szeliski book chapters 1-3 and optionally look over chapter 7.**

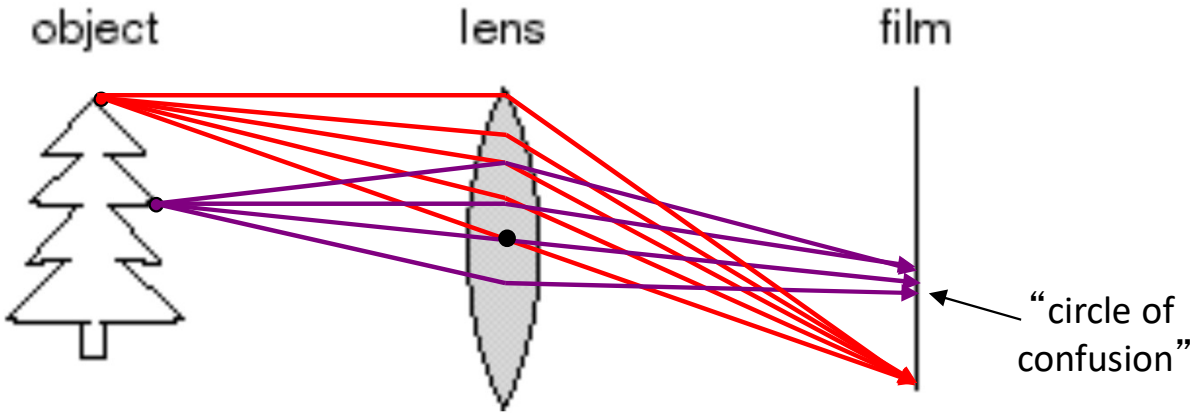
# Preview of next part of lecture

- Light transport, sensing, filtering, statistics of natural images

- Filtering study questions:
  - What should the mean of a filter be? Why?
  - What happens to filter response if you scale an input image?
  - What filter can respond to step edges, blobs?
  - What filter responds to an arbitrary pattern?
  - How can you implement a filter as a matrix multiply? (This matters for GPUs)
  - What size is the response of an image to a filter?
  - What happens if you apply one filter then another?
  - What happens if you apply a filter many times?
  - What is the computational complexity of applying a filter?

Will put out an assignment to implement and analyze the “dead leaves” model of image statistics.

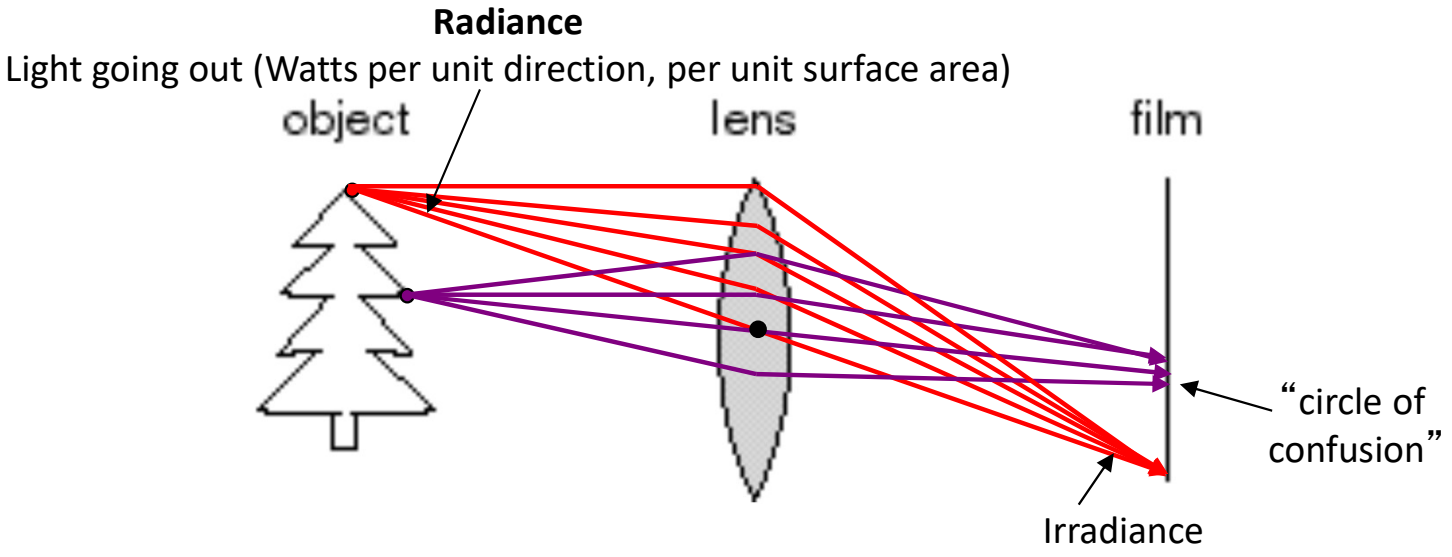
# Adding a lens




- A lens focuses light onto the film
  - There is a specific distance at which objects are “in focus”
  - Other points are blurred out over the film

Slide by Steve Seitz

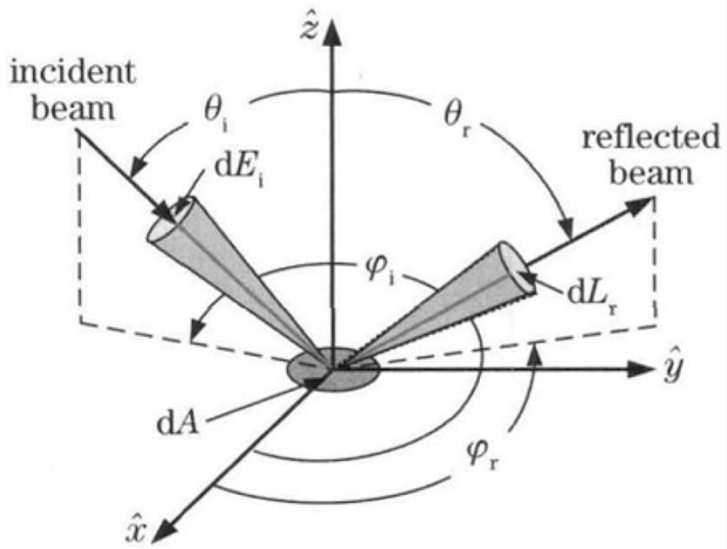
# Adding a lens




- A lens focuses light onto the film
  - There is a specific distance at which objects are “in focus”
  - Other points are blurred out over the film

# Bidirectional reflectance distribution function
# BRDF

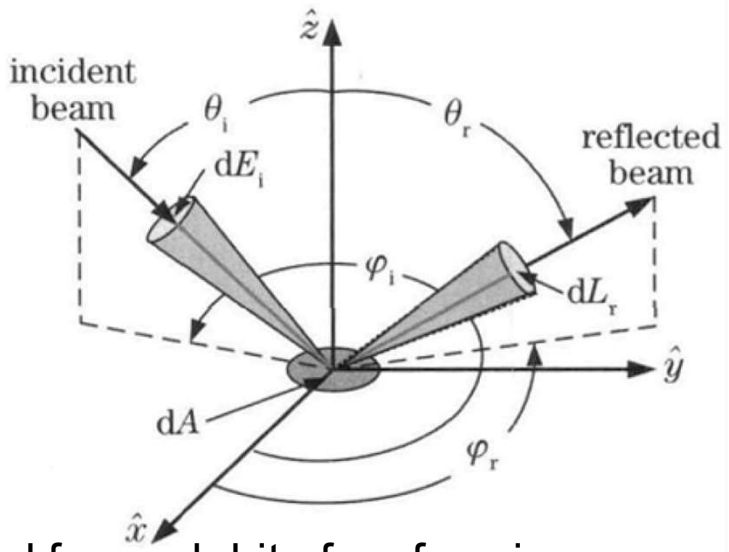
**Irradiance**
Light coming in
Watts per unit surface area




**Radiance**
Light going out
Watts per unit direction, per unit surface area

# Bidirectional reflectance distribution function BRDF

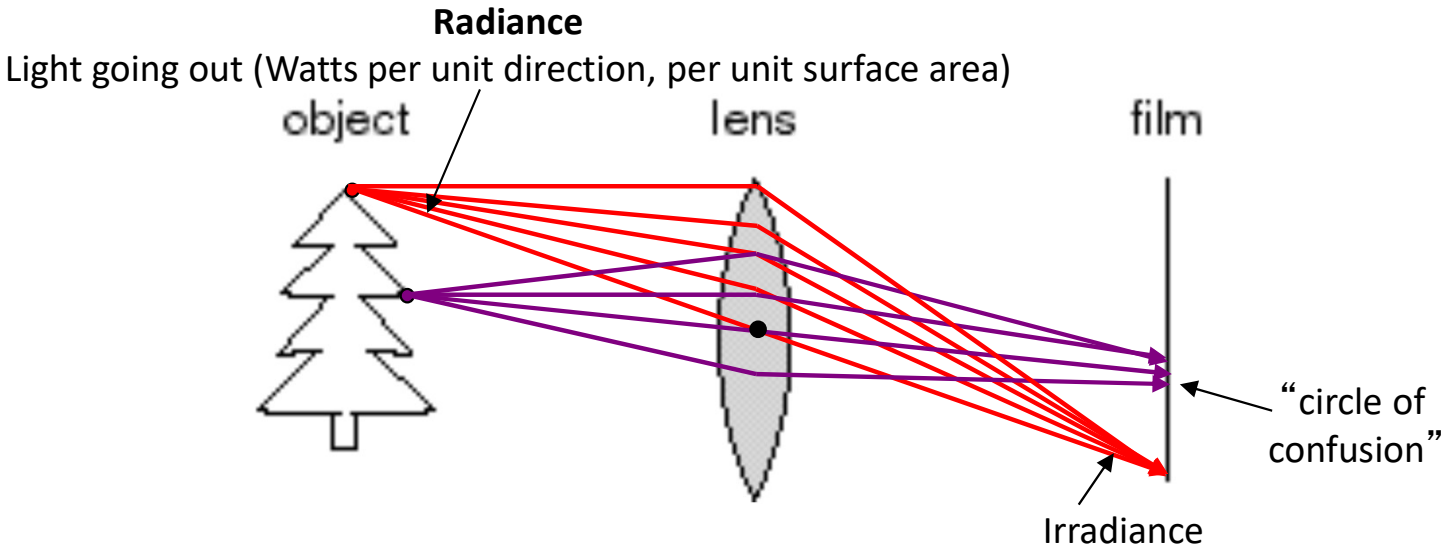
**Irradiance**
Light coming in
Watts per unit surface area




**Radiance**
Light going out
Watts per unit direction, per unit surface area

Can write an integral for each bit of surface in a scene, and make equalities for light going out and then going into another point (and for light emitted and absorbed). This is called “global illumination” as it accounts for all light, not just direct illumination from a light source.

# Adding a lens




- A lens focuses light onto the film
  - There is a specific distance at which objects are “in focus”
  - Other points are blurred out over the film

# Photons -> electrons -> numbers



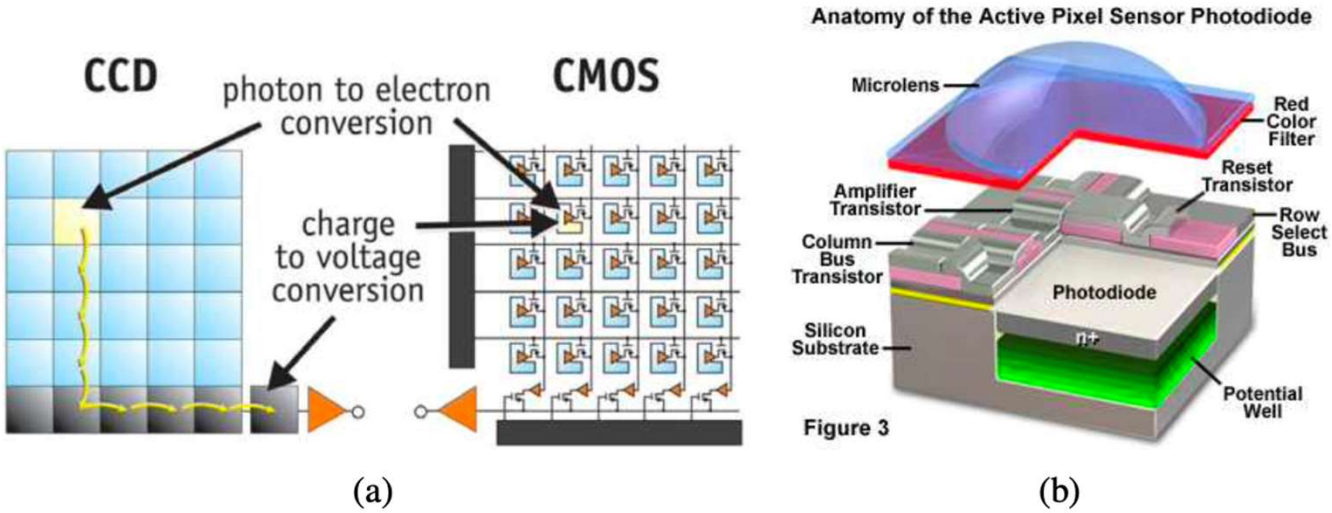



**Figure 2.24** *Digital imaging sensors: (a) CCDs move photogenerated charge from pixel to pixel and convert it to voltage at the output node; CMOS imagers convert charge to voltage inside each pixel (Litwiller 2005) © 2005 Photonics Spectra; (b) cutaway diagram of a CMOS pixel sensor, from https://micro.magnet.fsu.edu/primer/digitalimaging/cmosimagesensors.html.*

From Szeliski book

European Conference on Computer Vision
↳ ECCV 2002: **Computer Vision — ECCV 2002** pp 158–172 | Cite as

Optional reading on pedagogy of features

# Image Processing Done Right

Jan J. Koenderink & Andrea J. van Doorn

Conference paper | First Online: 01 January 2002
**3673** Accesses | **29** Citations | **1** Altmetric

Part of the Lecture Notes in Computer Science book series (LNCS,volume 2350)

## Abstract

A large part of "image processing" involves the computation of significant points, curves and areas ("features"). These can be defined as loci where absolute differential invariants of the image assume fiducial values, taking spatial scale and intensity (in a generic sense) scale into account. "Differential invariance" implies a group of "similarities" or "congruences". These "motions" define the geometrical structure of image space. Classical Euclidian invariants don't apply to images because image space is non-Euclidian. We analyze image structure from first principles and construct the fundamental group of image space motions. Image space is a Cayley-Klein geometry with one isotropic dimension. The analysis leads to a principled definition of "features" and the operators that define them.

"Intensity" is a generic name for a flux, the amount of stuff collected within a certain aperture centered at a certain location. In the case of a CCD array the aperture is set by the sensitive area and the flux is proportional with the number of absorbed photons collected in a given time window. One treats this as the continuous distribution of some "density", in the case of the CCD chip the number of absorbed photons per unit area per unit time, that is the irradiance. This goes beyond the observable and is often inadvisable because the "stuff" may be granular at the microscale. In the case of the CCD chip the grain is set by the photon shot noise. It is also inadvisable because natural images fail to be "nice" functions of time and place when one doesn't "tame" them via a finite collecting aperture or "inner scale". Only such tamed images are observable[4], this means that any image should come with an inner scale. This often is nilly willy the case. For instance, in the case of the CCD chip the inner scale is set by the size of its photosensitive elements. But no one stops you from changing the inner scale artificially. When possible this is a boon, because it rids one of the artificial pixelation. "Pixel fu[redacted]" is in a different ballpark from image processing proper, though it sometimes is a necessary evil due to real world constraints. Because of a number of technical reasons the preferred way to set the inner scale is to use Gaussian smoothing[4]. Here we assume that the "intensity" z(x, y) is a smooth function of the Cartesian coordinates {x, y} of the picture plane with a well defined inner scale. We assume that the intensity is positive definite throughout.

Filtering and convolution from Szeliski book

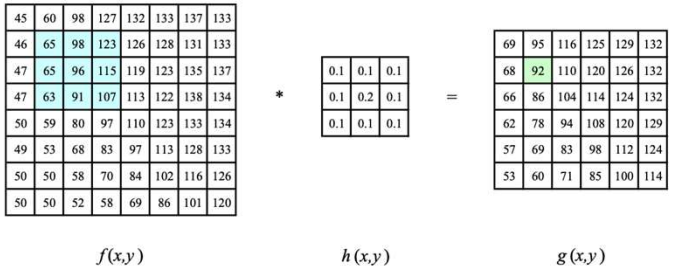



**Figure 3.10** *Neighborhood filtering (convolution): The image on the left is convolved with the filter in the middle to yield the image on the right. The light blue pixels indicate the source neighborhood for the light green destination pixel.*

we look at non-linear operators such as morphological filters and distance transforms.

The most widely used type of neighborhood operator is a *linear filter*, where an output pixel's value is a weighted sum of pixel values within a small neighborhood $\mathcal{N}$ (Figure 3.10),

$$g(i,j) = \sum_{k,l} f(i+k, j+l)h(k,l). \tag{3.12}$$

The entries in the weight *kernel* or *mask* $h(k,l)$ are often called the *filter coefficients*. The above *correlation* operator can be more compactly notated as

$$g = f \otimes h. \tag{3.13}$$

A common variant on this formula is

$$g(i,j) = \sum_{k,l} f(i-k, j-l)h(k,l) = \sum_{k,l} f(k,l)h(i-k, j-l), \tag{3.14}$$

where the sign of the offsets in $f$ has been reversed, This is called the *convolution* operator,

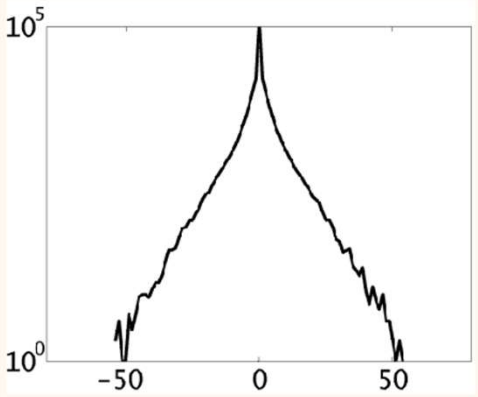
$$g = f * h, \tag{3.15}$$

and $h$ is then called the *impulse response function*.[5] The reason for this name is that the kernel function, $h$, convolved with an impulse signal, $\delta(i,j)$ (an image that is 0 everywhere except at the origin) reproduces itself, $h * \delta = h$, whereas correlation produces the reflected signal. (Try this yourself to verify that it is so.)

# Statistics of filter responses in natural images

Natural images have an interesting behavior when represented using wavelets or, in general, when seen through a set of orientation and scale selective band-pass filters. **First**, typically, most responses of the filters (corresponding to uniform soft texture areas - gray areas of the subband below) have a close to zero value, whereas a few of them (corresponding to the responses to edges, lines, corners and other localized salient features - black and white features on the image below, on the left panel) have comparatively very large amplitude responses. Therefore, if we look at the histogram of a subband of a natural image it typically has a **strong peak at zero** and long **heavy tails** (figure below, right panel):

**Left**: *A wavelet subband of a typical image*



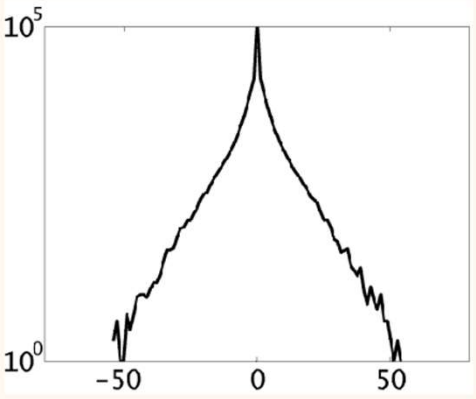

**Right**: *Typical subband histogram*

Note the logarithmic scale of the ordinates. This feature was already observed by [Field 1987] and first modeled by [Mallat 1989], using a generalized Gaussian (see also [Simoncelli 1996, Buccigrossi 1997]). It is often referred to as the "high kurtosis", "leptokurtosis behavior" or "**sparseness**" of the wavelet coefficients of a subband.

Slide from Simoncelli

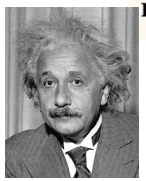
# Statistics of filter responses in natural images

Natural images have an interesting behavior when represented using wavelets or, in general, when seen through a set of orientation and scale selective band-pass filters. **First**, typically, most responses of the filters (corresponding to uniform soft texture areas - gray areas of the subband below) have a close to zero value, whereas a few of them (corresponding to the responses to edges, lines, corners and other localized salient features - black and white features on the image below, on the left panel) have comparatively very large amplitude responses. Therefore, if we look at the histogram of a subband of a natural image it typically has a **strong peak at zero** and long **heavy tails** (figure below, right panel):




**Left**: *A wavelet subband of a typical image* **Right**: *Typical subband histogram*

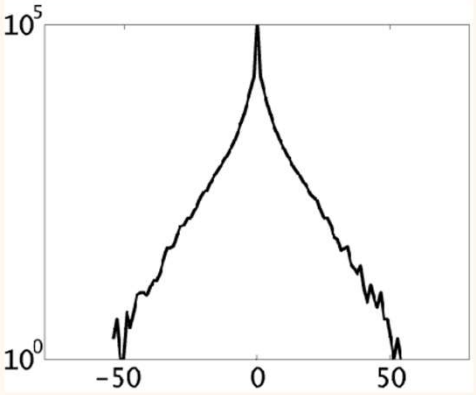
Note the logarithmic scale of the ordinates. This feature was already observed by [Field 1987] and first modeled by [Mallat 1989], using a generalized Gaussian (see also [Simoncelli 1996, Buccigrossi 1997]). It is often referred to as the "high kurtosis", "leptokurtosis behavior" or "**sparseness**" of the wavelet coefficients of a subband.



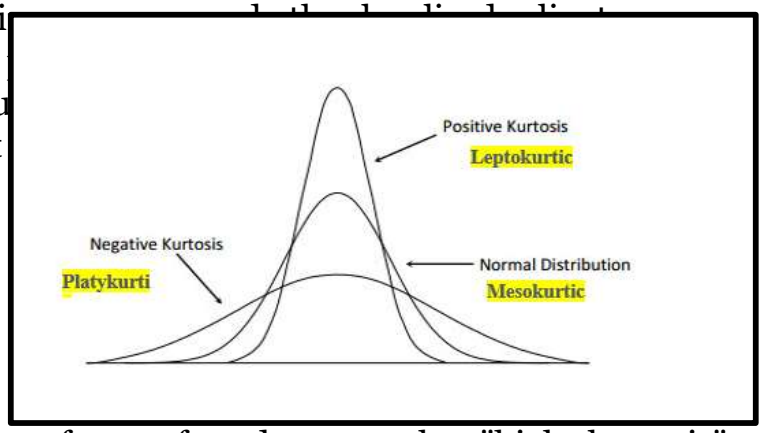
Slide from Simoncelli

# Statistics of filter responses in natural images

Natural images have an interesting behavior when represented using wavelets or, in general, when seen through a set of orientation and scale selective band-pass filters. **First**, typically, most responses of the filters (corresponding to uniform soft texture areas - gray areas of the subband below) have a close to zero value, whereas a few of them (corresponding to the responses to edges, li
features - black and white features on the image below, on the left
amplitude responses. Therefore, if we look at the histogram of a su
a **strong peak at zero** and long **heavy tails** (figure below, right




**Left**: *A wavelet subband of a typical image* **Right**: *Typical subband histogram*




often referred to as the "high kurtosis", "leptokurtosis behavior" or "**sparseness**" of the wavelet coefficients of a subband.

# “Dead leaves” model

Figure 1: Left : simulation of a dead leaves model, where the grain $X_0$ is a disk with constant radius; Right : simulation of a dead leaves model, where the grain $X_0$ is a disk with a uniformly distributed radius.

**The dead leaves model : general results and limits at small scales**

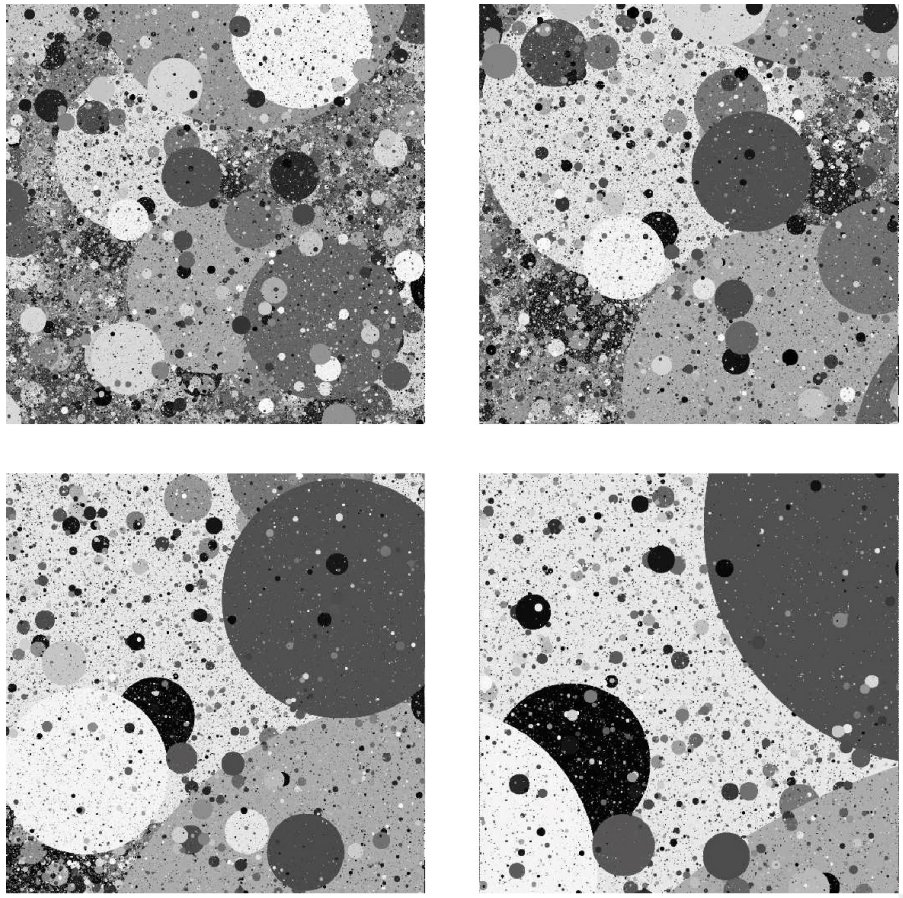
Y. Gousseau, F. Roueff • Published 1 December 2003 • Mathematics • arXiv: Probability

# “Dead leaves” model

**The dead leaves model : general results and limits at small scales**

Y. Gousseau, F. Roueff · Published 1 December 2003 · Mathematics · arXiv: Probability

# “Dead leaves” model

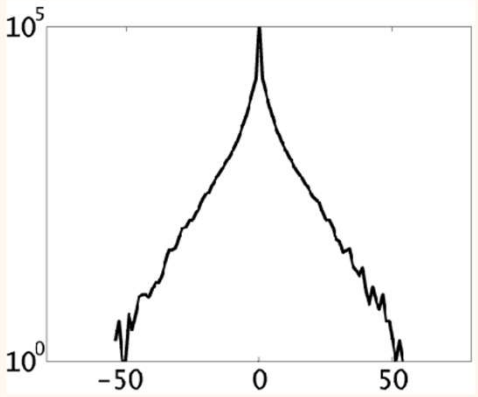


**The dead leaves model : general results and limits at small scales**

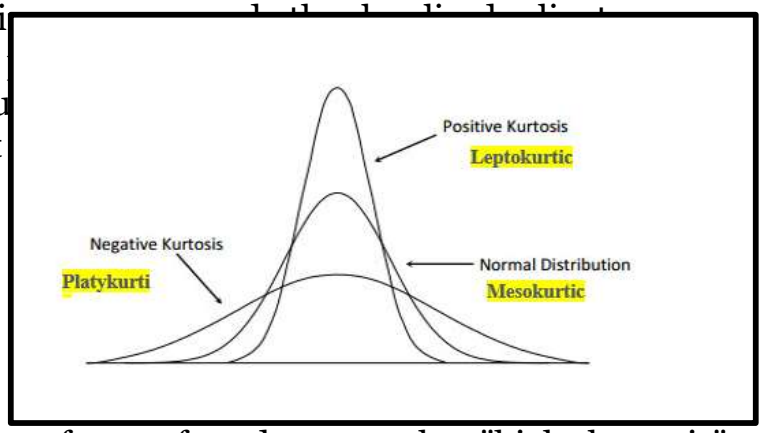
Y. Gousseau, F. Roueff · Published 1 December 2003 · Mathematics · arXiv: Probability

# Statistics of filter responses in natural images

Natural images have an interesting behavior when represented using wavelets or, in general, when seen through a set of orientation and scale selective band-pass filters. **First**, typically, most responses of the filters (corresponding to uniform soft texture areas - gray areas of the subband below) have a close to zero value, whereas a few of them (corresponding to the responses to edges, li
features - black and white features on the image below, on the left
amplitude responses. Therefore, if we look at the histogram of a su
a **strong peak at zero** and long **heavy tails** (figure below, right




**Left**: *A wavelet subband of a typical image* **Right**: *Typical subband histogram*




often referred to as the "high kurtosis", "leptokurtosis behavior" or "**sparseness**" of the wavelet coefficients of a subband.

**Assignment**: show this for linear filters on **collections** of **natural** images, also find **images and filters** where it is not true.

**Assignment**: show this (heavy tail) for linear filters on **collections** of **natural** images, also find **images and filters** where it is not true.

1. Find some images natural and not
2. Write some code to perform linear filtering
3. Demonstrate that this works (make a figure with an image and the filter response)
4. Collect statistics of filter responses – make a plot of log of histogram
    1. A filter on an image
    2. A set of filters on an image
    3. A filter on a set of images
    4. A set of filters on a set of images
5. Find images/filters where it is not true, show this in a figure
6. Include code and brief writeup with figures
7. Extra: (When) Can you replicate these results with a dead leaves model?
8. Due before next class on Canvas (turn in a pdf showing your notebook with code, figures, and explanation

# Recap of part 2 of lecture

- Light transport irradiance, brdf, radiance, big integral.
- Converting to numbers
- Pixel f****, aliasing, and other problems
- Filtering
- Statistics of filter responses
- Dead leaves
- Assignment show distribution of filter responses in natural images.
- Questions?