Computational Approaches to Cameras

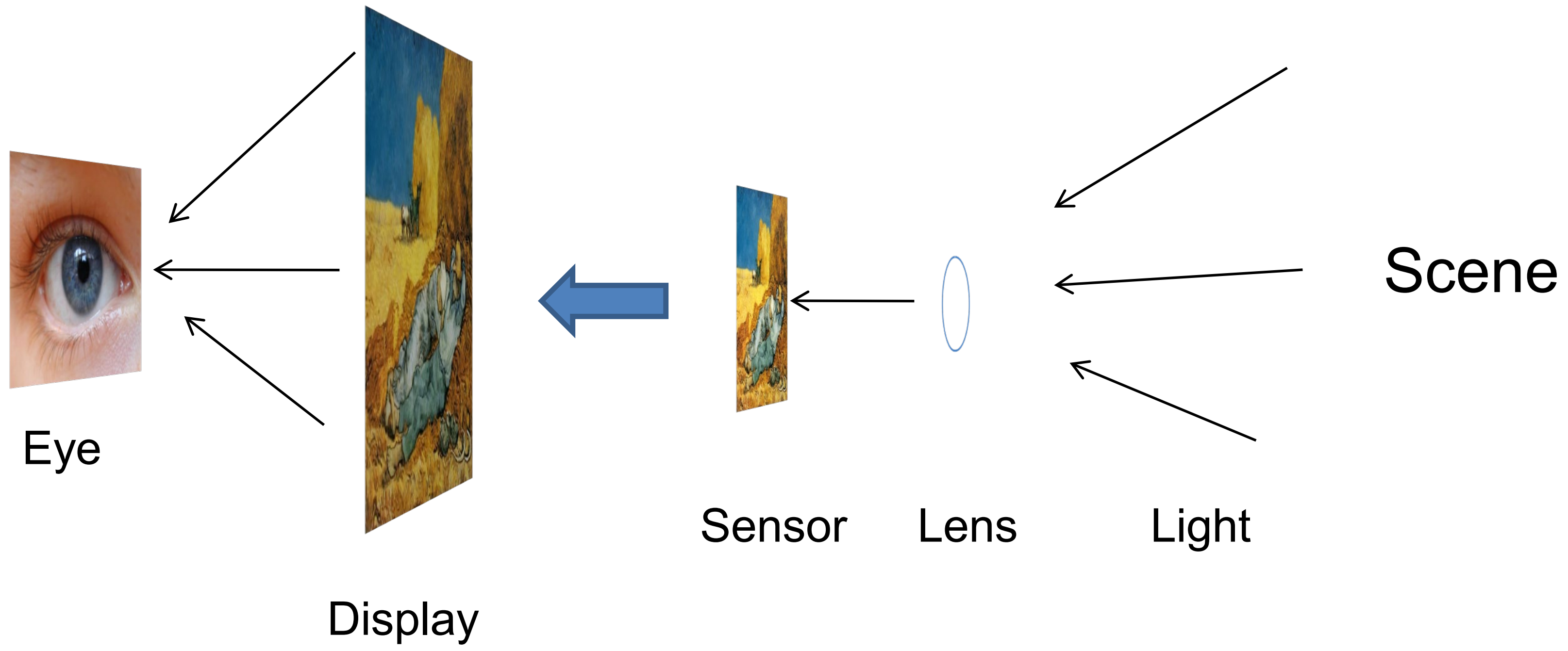
Magritte , *The False Mirror* (1935)

Computational Photography

Derek Hoiem, University of Illinois

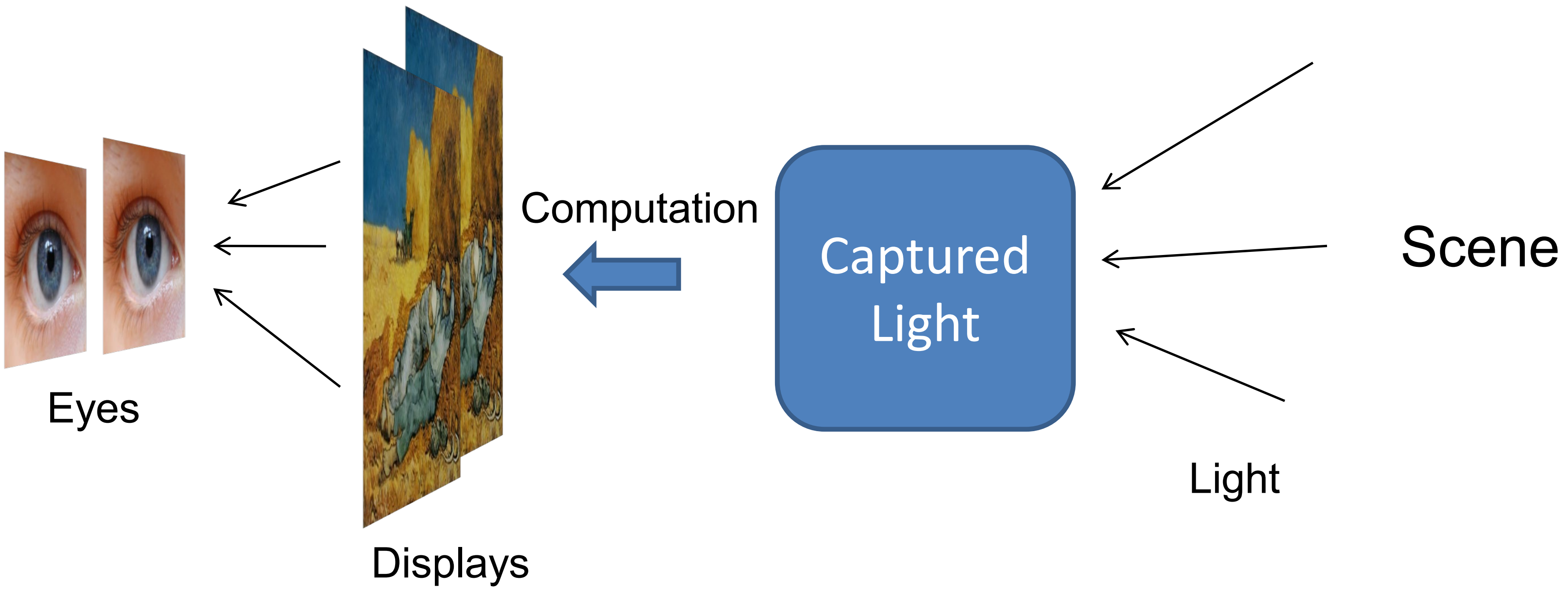
Conventional cameras

Conventional cameras are designed to capture light in a medium that is directly viewable



Computational cameras

With a computational approach, we can capture light and then figure out what to do with it



Questions for today

- How can we represent all of the information contained in light?
- What are the fundamental limitations of cameras?
- What sacrifices have we made in conventional cameras? For what benefits?
- How else can we design cameras for better focus, deblurring, multiple views, depth, etc.?

Representing Light: The Plenoptic Function

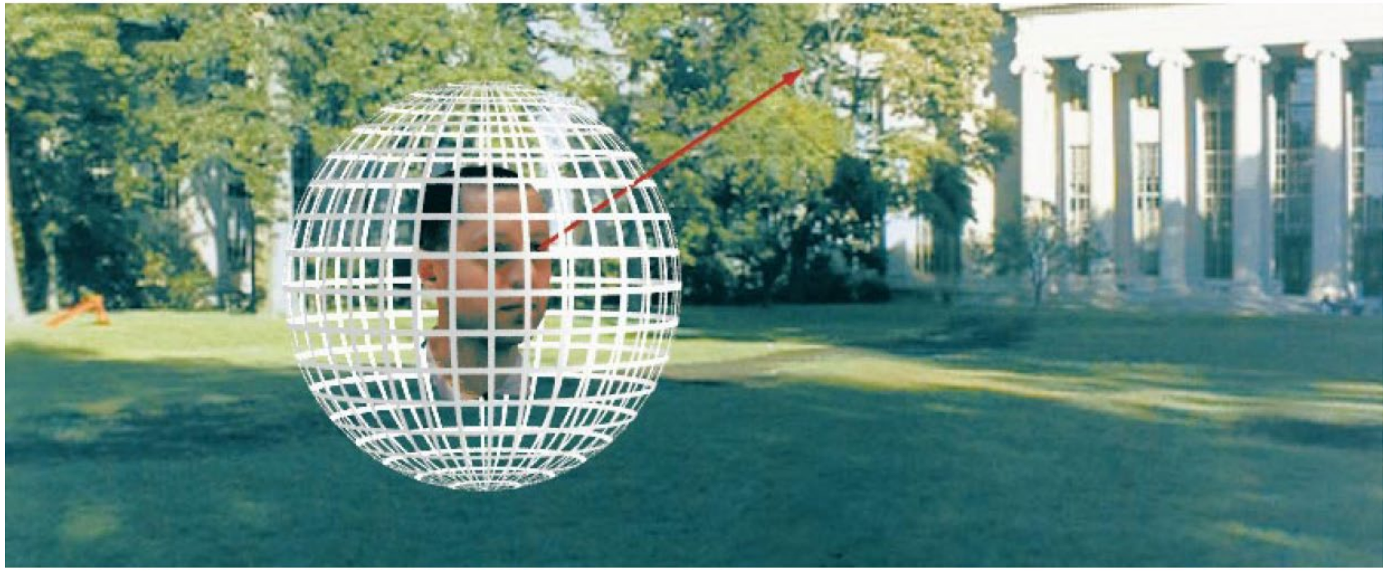


Figure by Leonard McMillan

Q: What is the set of all things that we can ever see?

A: The Plenoptic Function (Adelson & Bergen)

Let's start with a stationary person and try to parameterize <u>everything</u> that he can see…

Slides from Efros

Grayscale snapshot

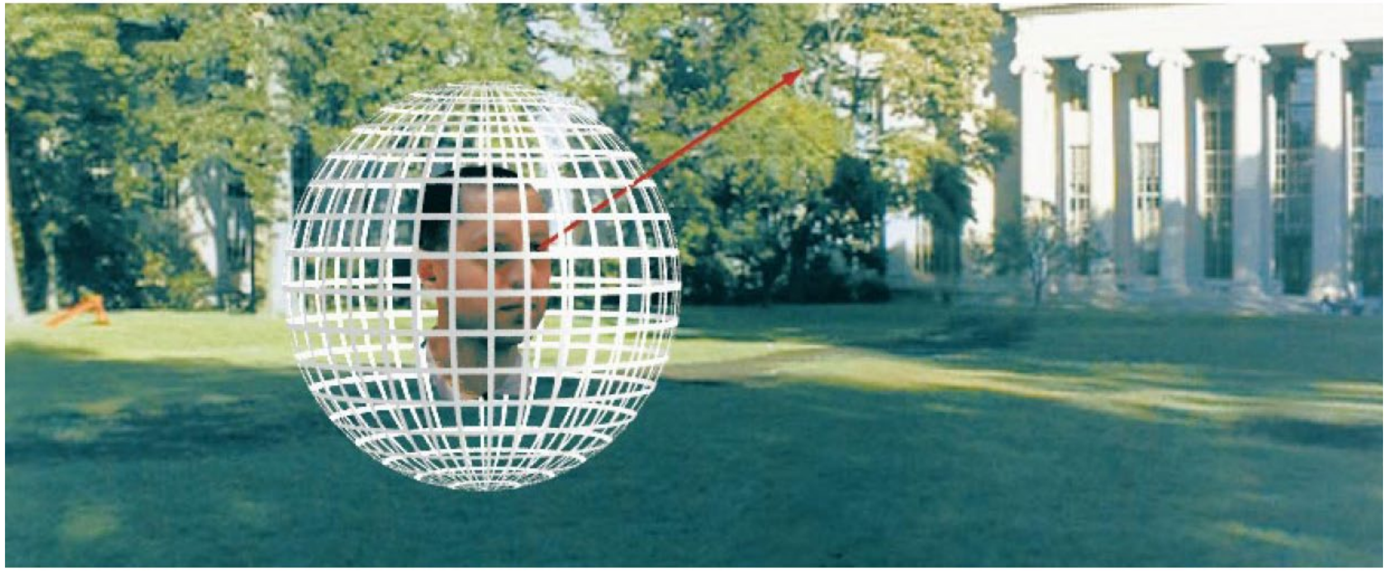
$$P(\theta,\phi)$$

is intensity of light

- Seen from a single view point
- At a single time
- Averaged over the wavelengths of the visible spectrum

(can also do *P(x,y),* but spherical coordinate are nicer)

Color snapshot

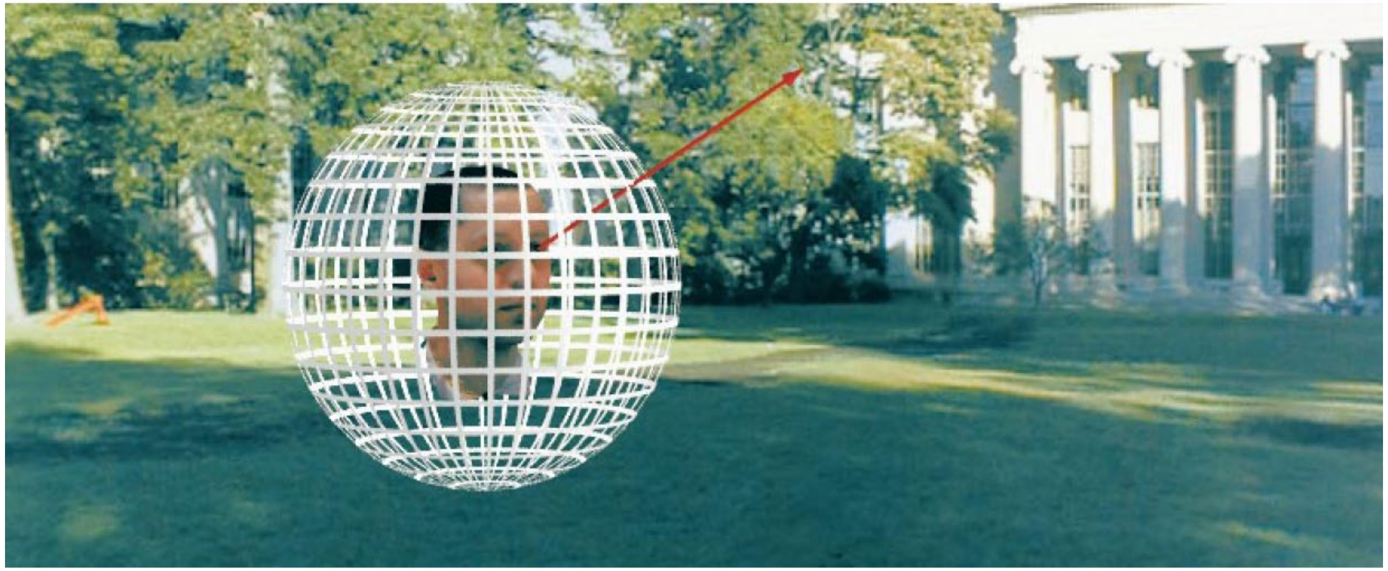


$$P(\theta,\phi,\lambda)$$

is intensity of light

- Seen from a single view point
- At a single time
- As a function of wavelength

A movie

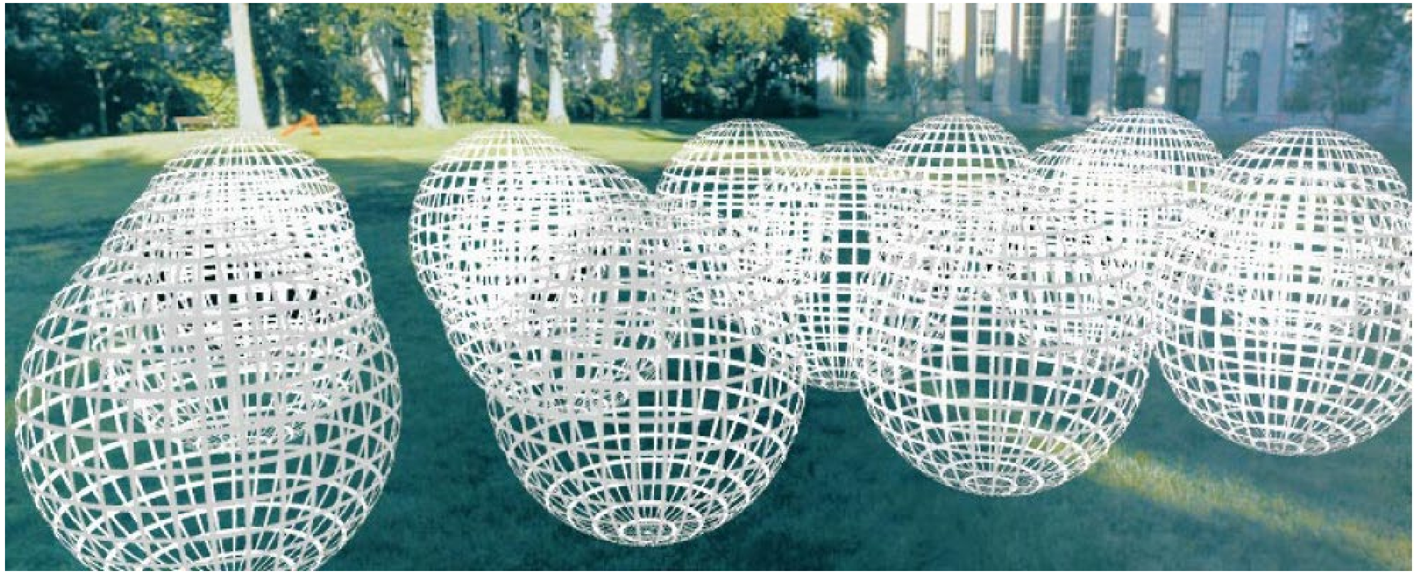


$$P(\theta,\phi,\lambda,t)$$

is intensity of light

- Seen from a single view point
- Over time
- As a function of wavelength

Holographic movie

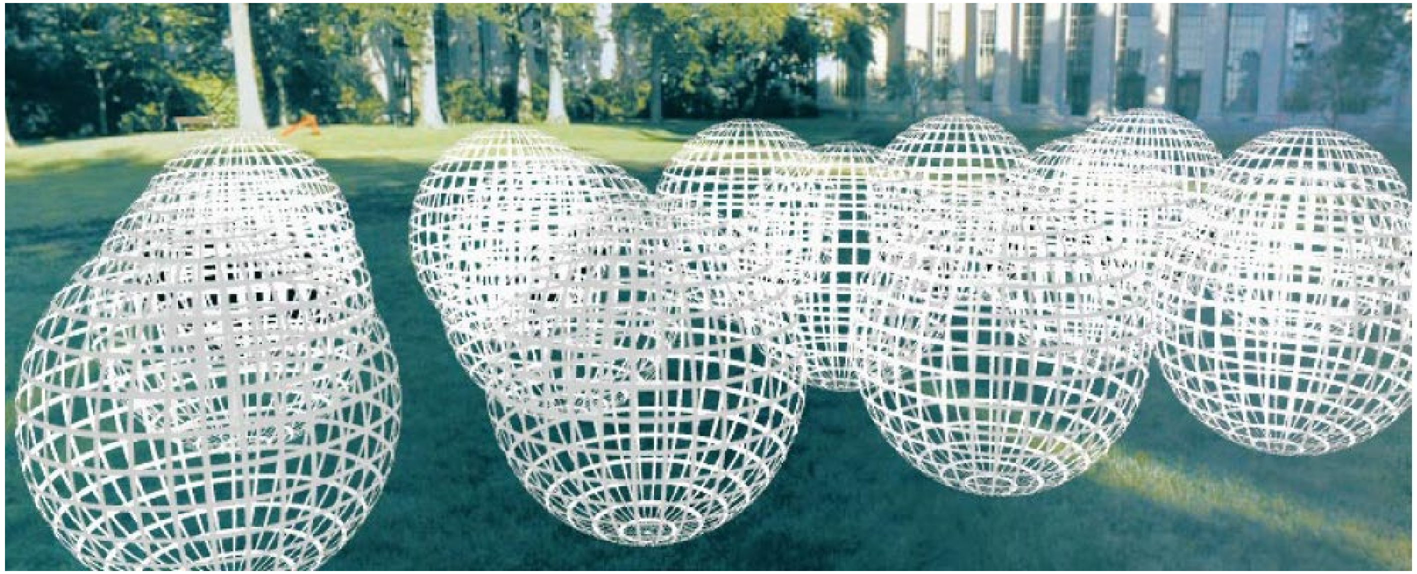


$$P(\theta,\phi,\lambda,t,V_X,V_Y,V_Z)$$

is intensity of light

- Seen from ANY viewpoint
- Over time
- As a function of wavelength

The Plenoptic Function



$$P(\theta,\phi,\lambda,t,V_X,V_Y,V_Z)$$

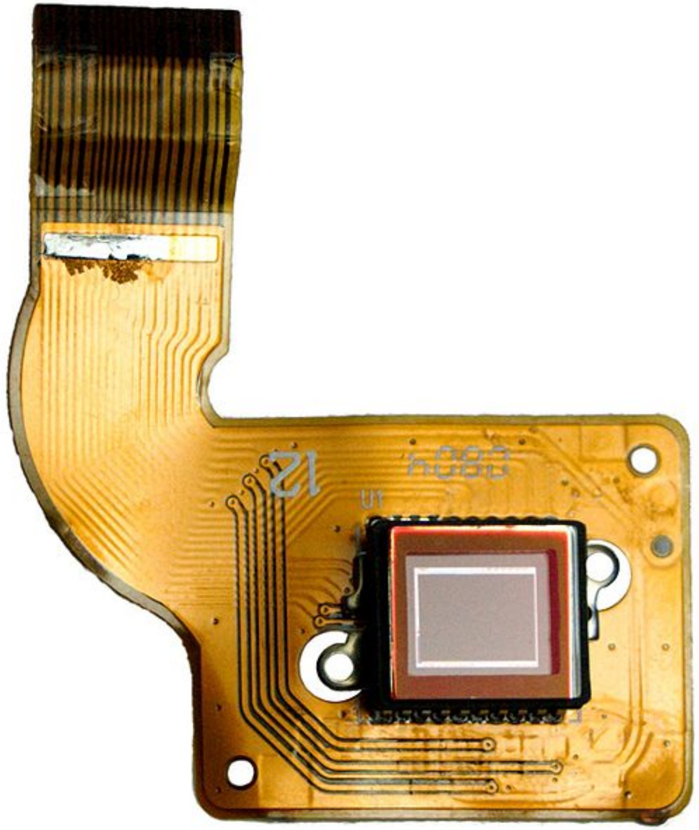
- Can reconstruct every possible view, at every moment, from every position, at every wavelength
- Contains every photograph, every movie, everything that anyone has ever seen!

Representing light

The atomic element of light: ~~a pixel~~ **a ray**

Fundamental limitations and trade-offs

- Only so much light in a given area to capture
- Basic sensor accumulates light at a set of positions from all orientations, over all time
- We want **intensity** of light at **a given time** at **one position** for a **set of orientations**
- Solutions:
  - funnel, constrain, redirect light
  - change the sensor



CCD inside camera

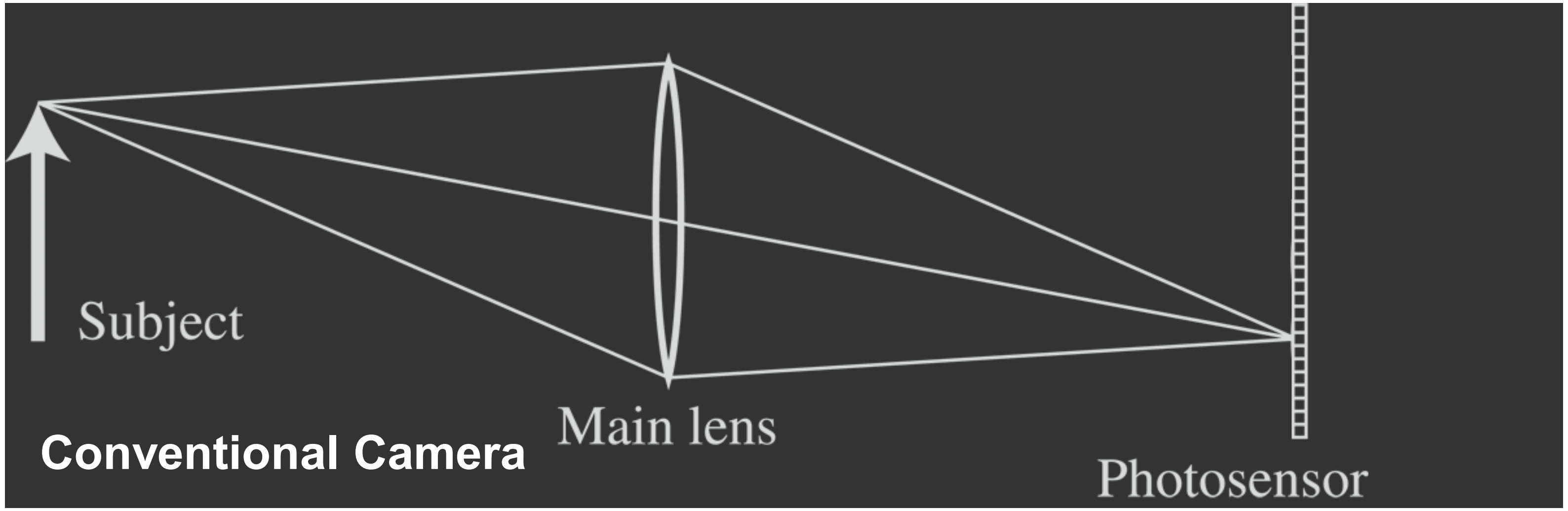
Trade-offs of conventional camera

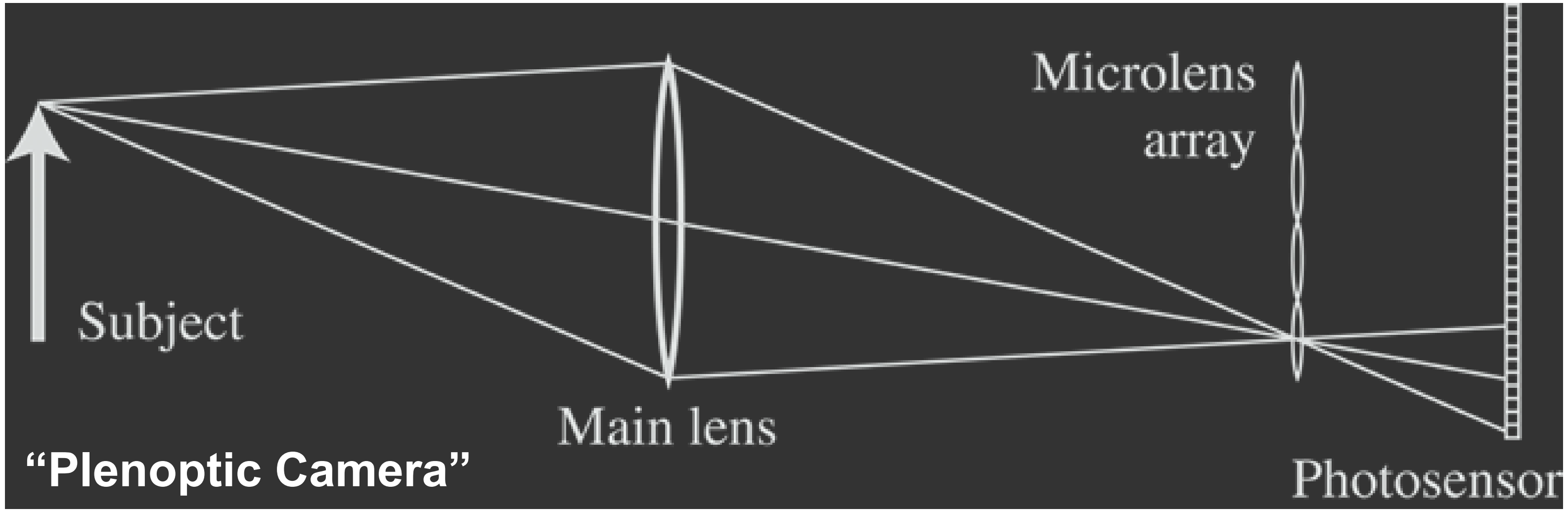
- Add a pinhole
  - ✓ Pixels correspond to small range of orientations at the camera center, instead of all gathered light at one position
  - ✗ Much less light hits sensor
- Add a lens
  - ✓ More light hits sensor
  - ✗ Limited depth of field
  - ✗ Chromatic aberration
- Add a shutter
  - Capture average intensity at a particular range of times
- Increase sensor resolution
  - ✓ Each pixel represents a smaller range of orientations
  - ✗ Less light per pixel
- Controls: aperture size, focal length, shutter time

How else can we design cameras?

What do they sacrifice/gain?

1. Light Field Photography with “Plenoptic Camera”



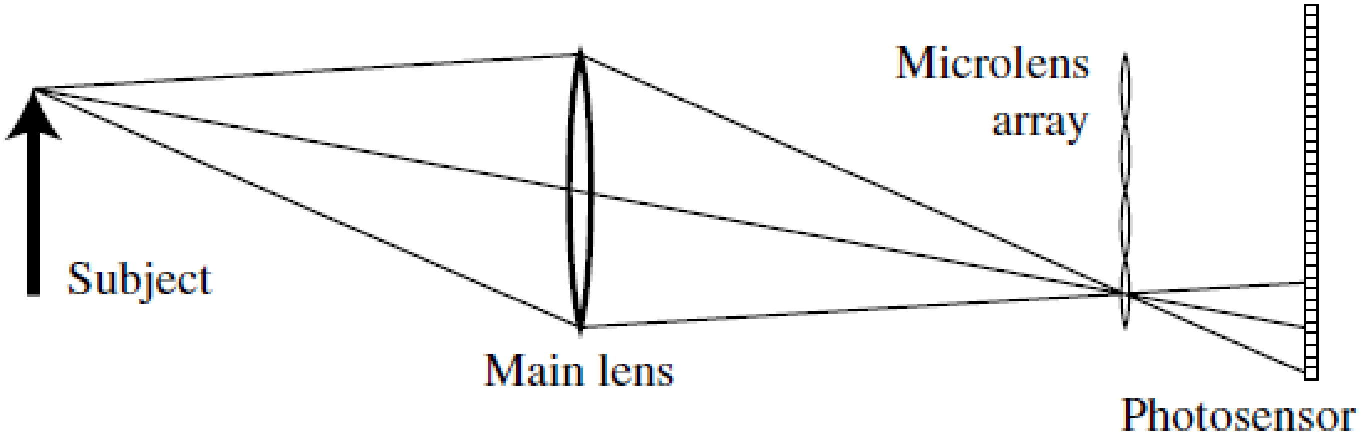


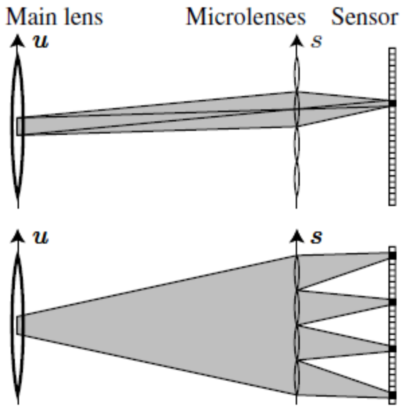
Adelson and Wang 1992

Ng et al. Stanford TR, 2005

Light field photography

- Like replacing the human retina with an insect compound eye
- Records where light ray hits the lens

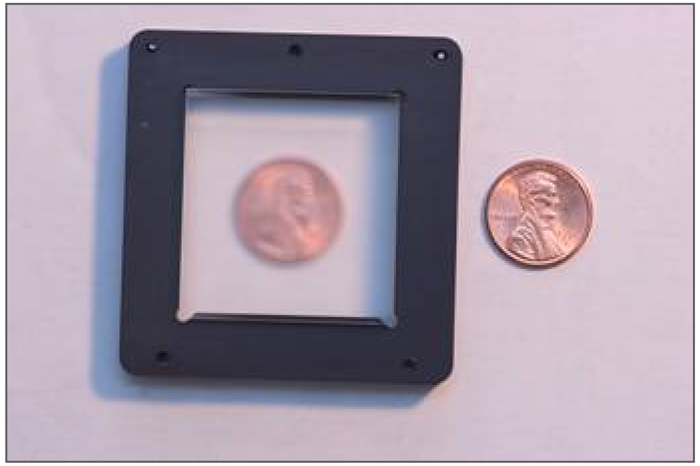


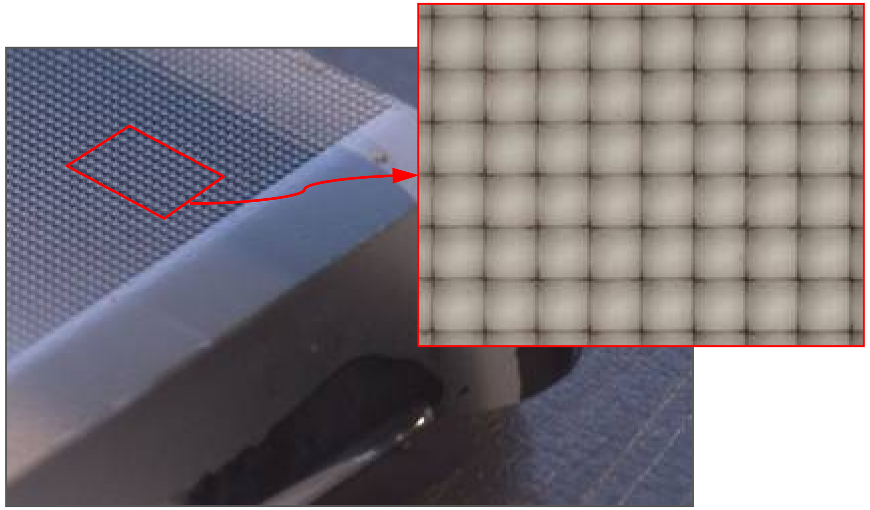


Stanford Plenoptic Camera [Ng et al 2005]

Contax medium format camera

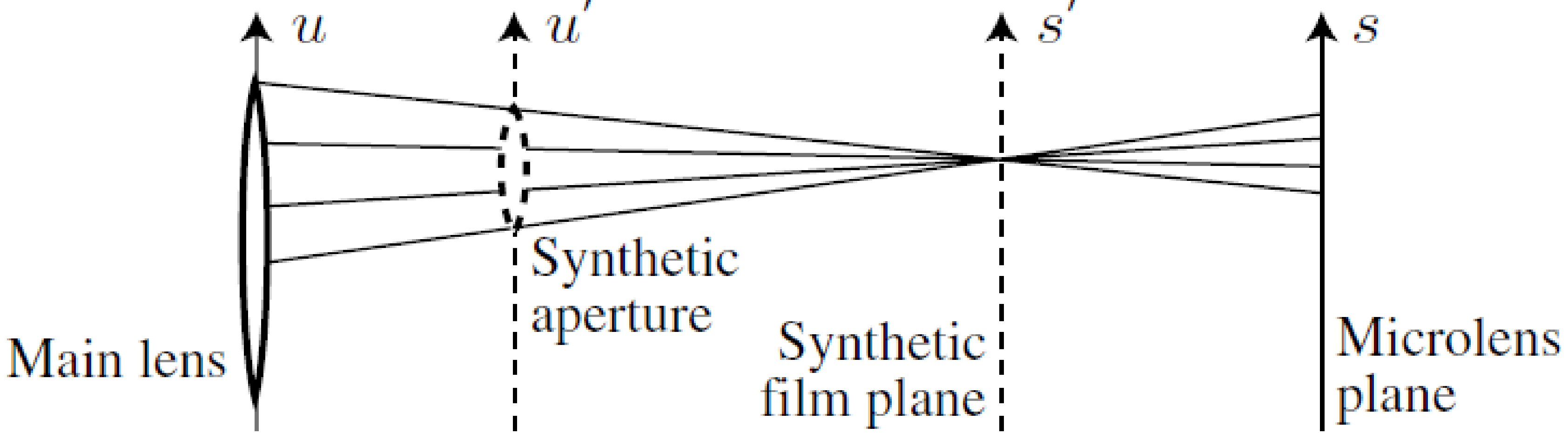

Kodak 16-megapixel sensor


Adaptive Optics microlens array

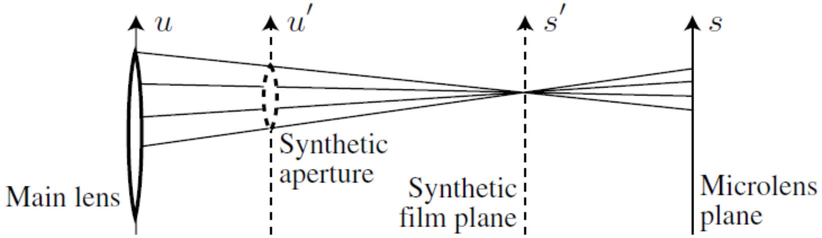

125μ square-sided microlenses

4000 × 4000 pixels ÷ 292 × 292 lenses = 14 × 14 pixels per lens

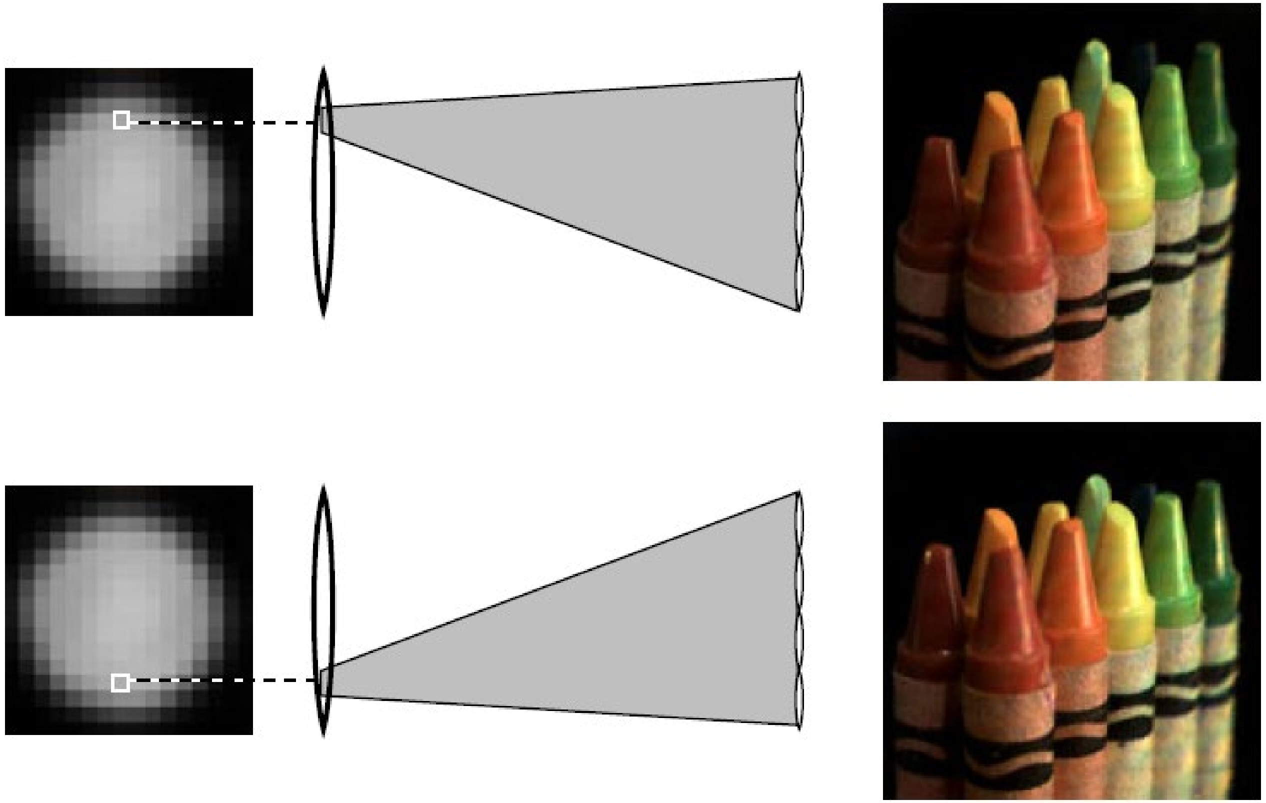
Light field photography: applications



Light field photography: applications

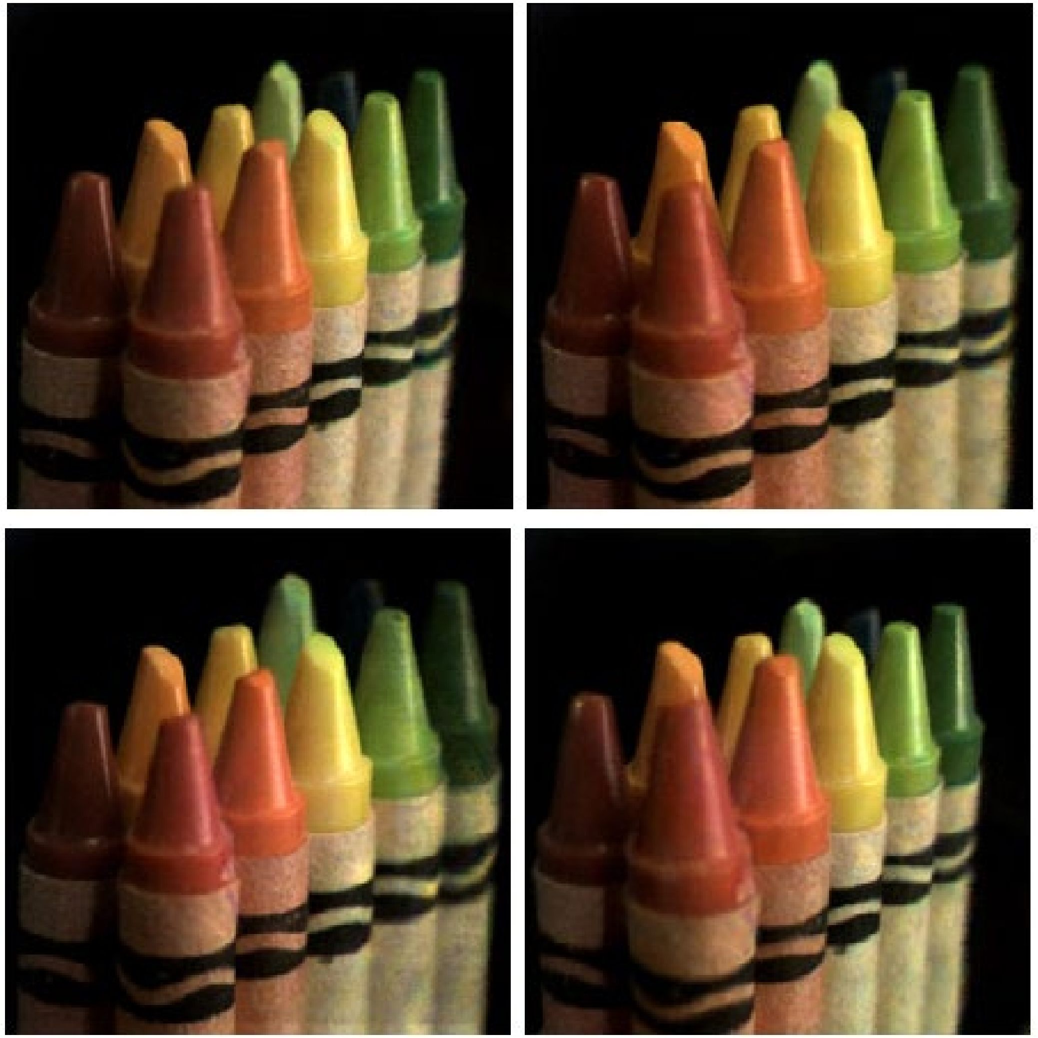


Change in viewpoint



Light field photography: applications

Change in viewpoint



Lateral

Along Optical Axis

Digital Refocusing

Light field photography w/ microlenses

- We gain
  - Ability to refocus or increase depth of field
  - Ability for small viewpoint shifts

- What do we lose (vs. conventional camera)?

2. Coded apertures

Image and Depth from a Conventional Camera with a Coded Aperture

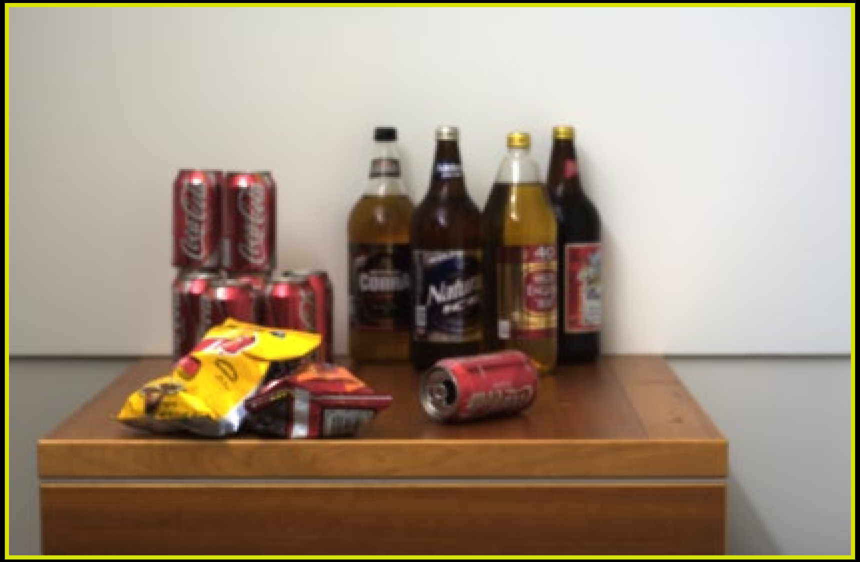
Anat Levin, Rob Fergus,
Frédo Durand, William Freeman

MIT CSAIL

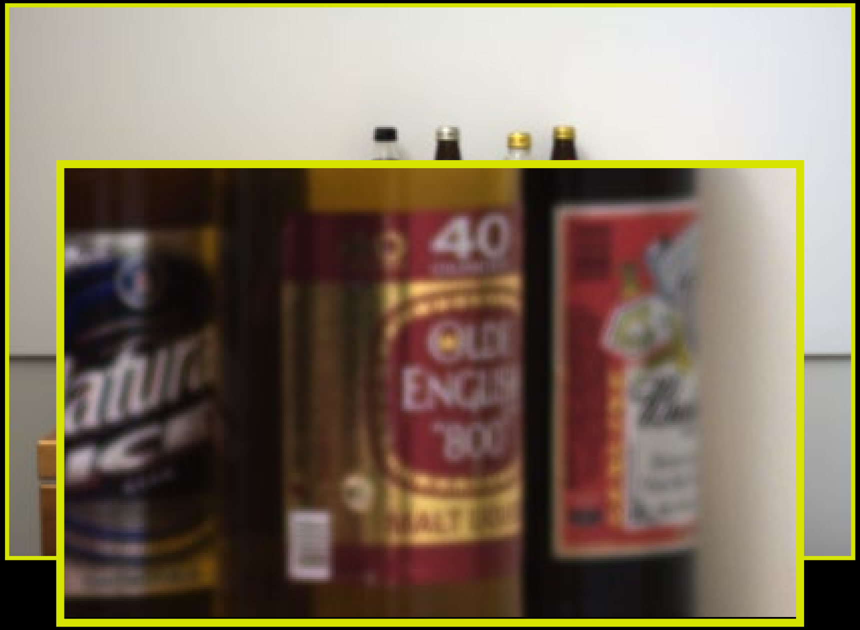
Slides from SIGGRAPH Presentation

Output #1: Depth map

Single input image:

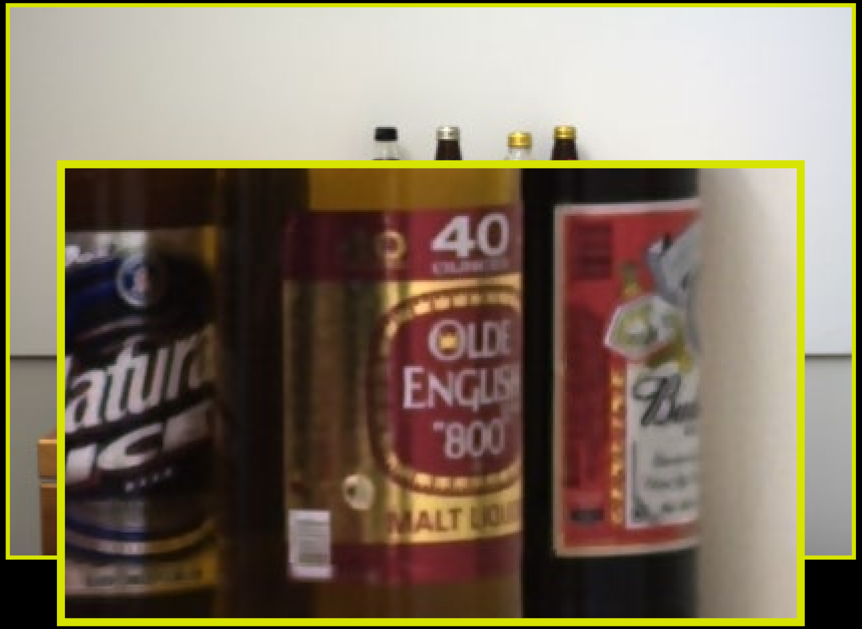


Single input image:



Output #1: Depth map

Output #2: All-focused image



Lens and defocus

Lens’ aperture

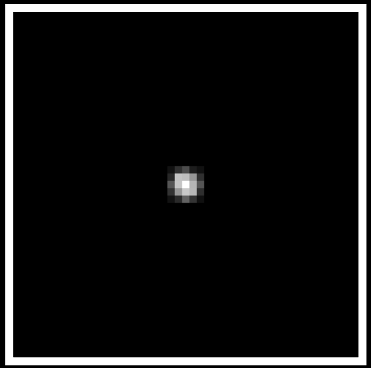
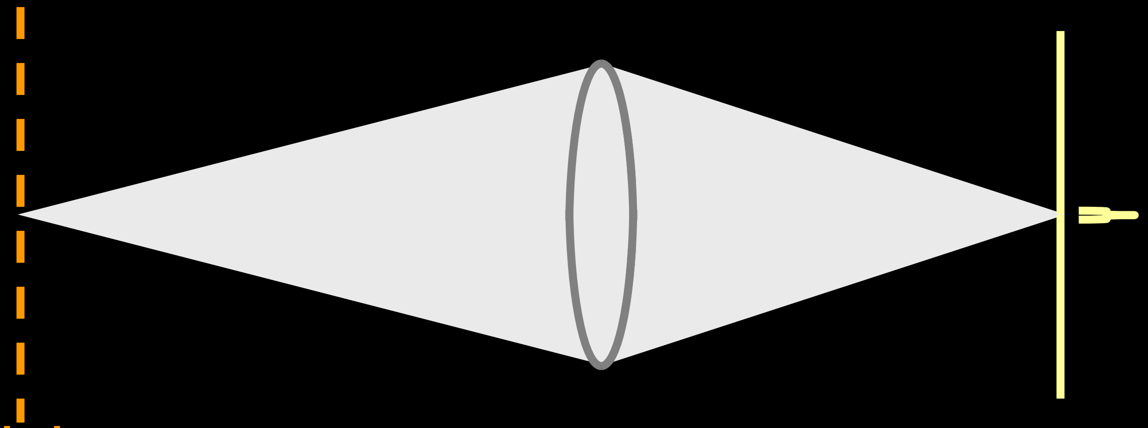


Image of a point light source

Lens

Camera sensor



Point spread function

Focal plane

Lens and defocus

Lens' aperture

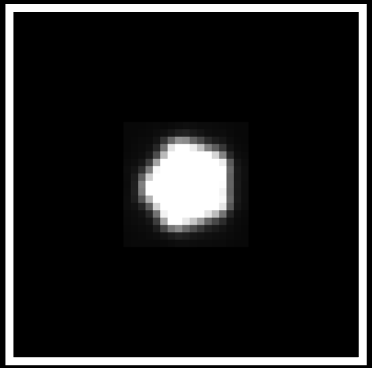
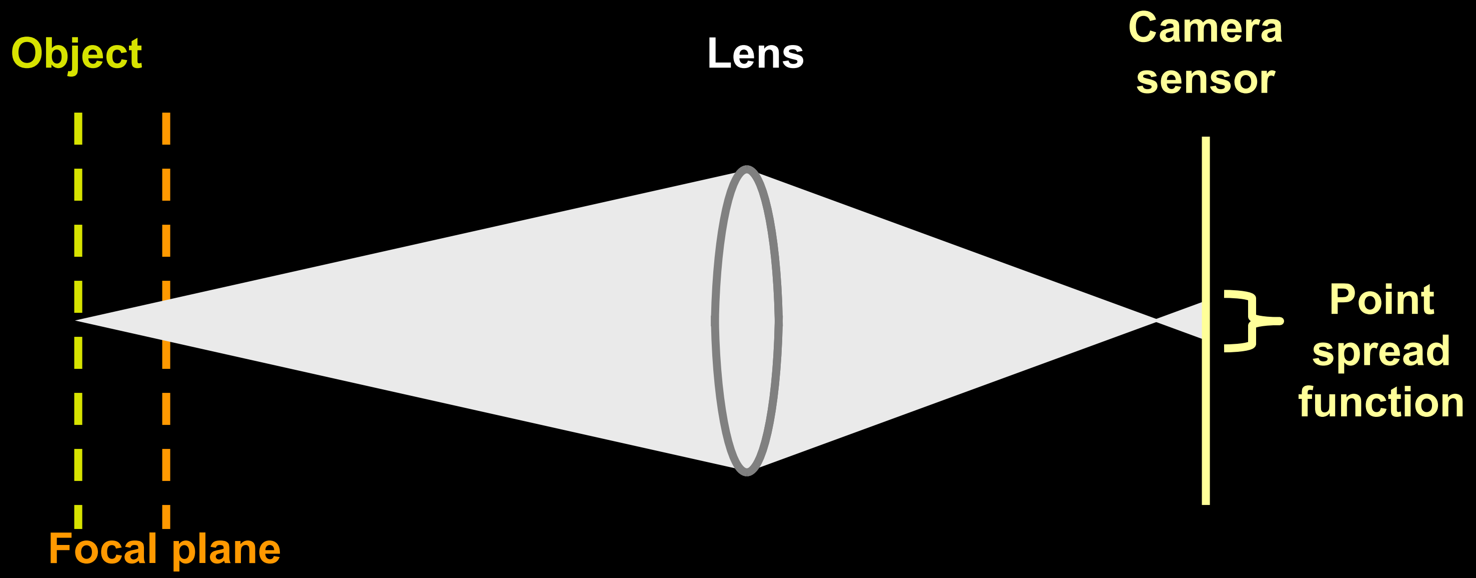


Image of a defocused point light source



Lens and defocus

Lens' aperture

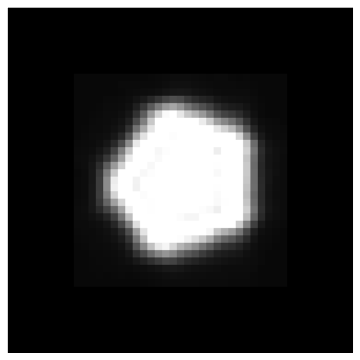
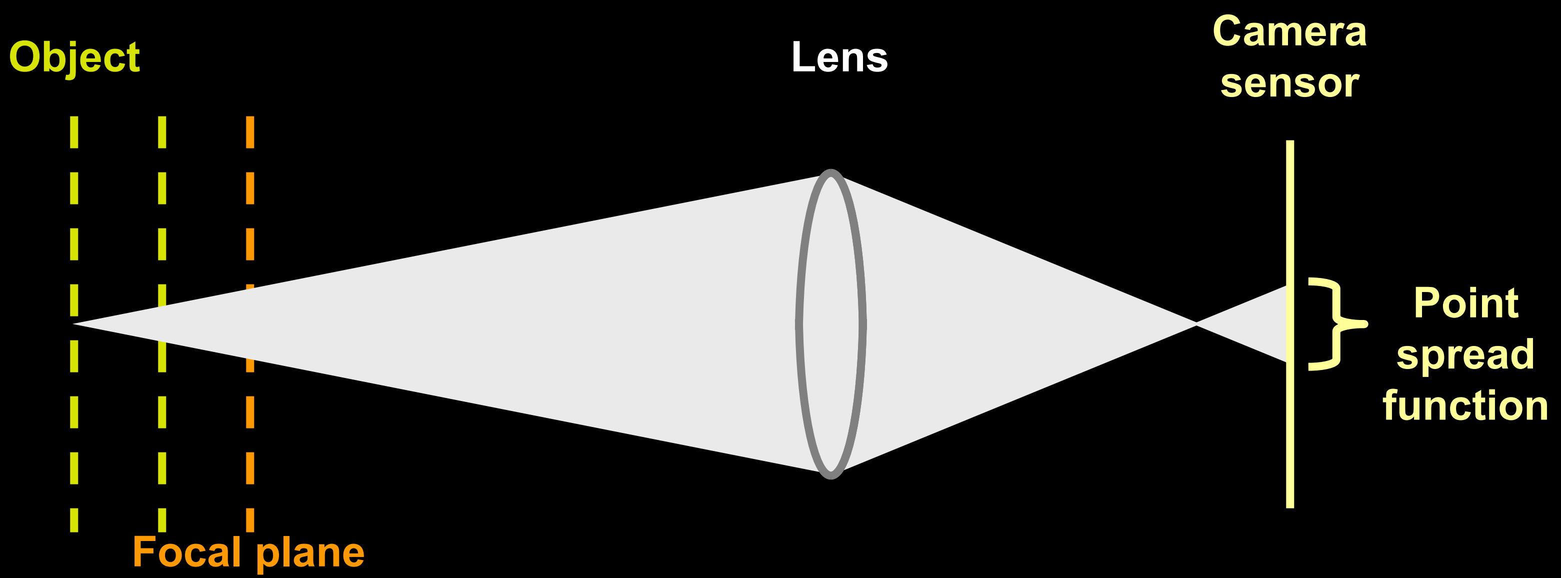


Image of a defocused point light source



Lens and defocus

Lens’ aperture

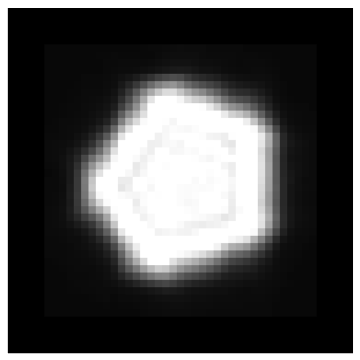
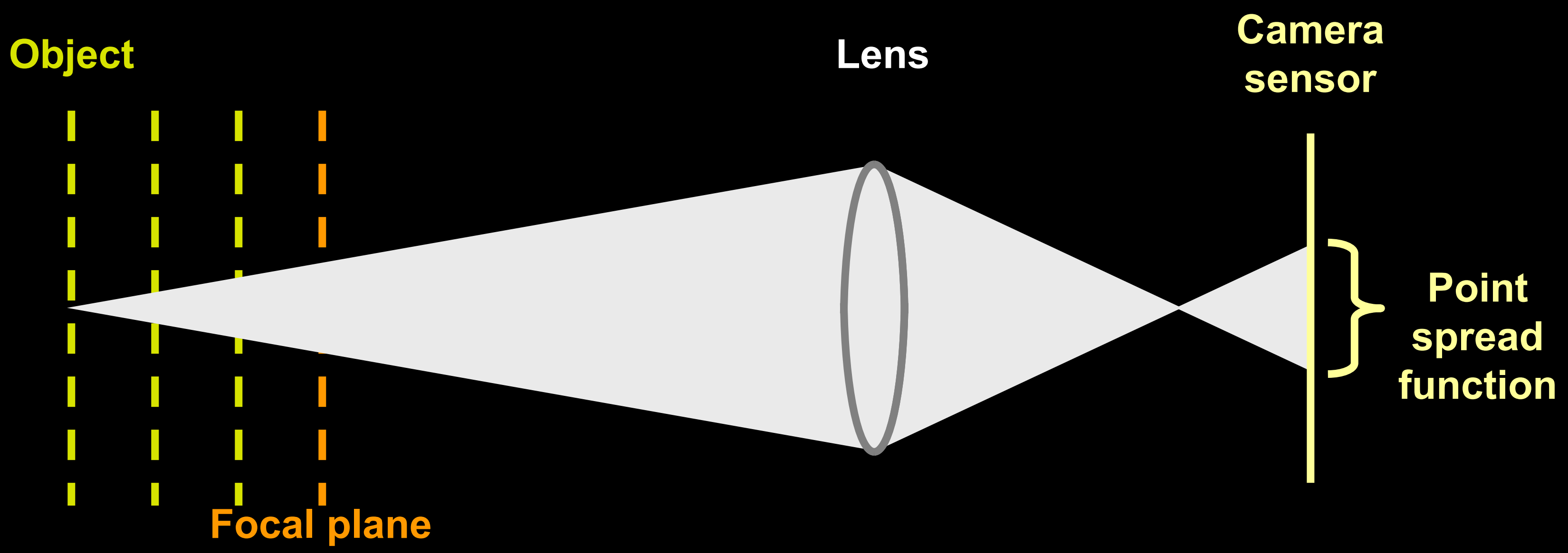


Image of a defocused point light source



Lens and defocus

Lens' aperture

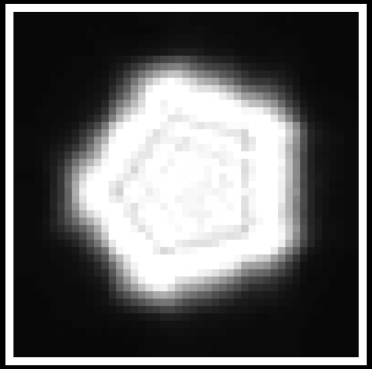
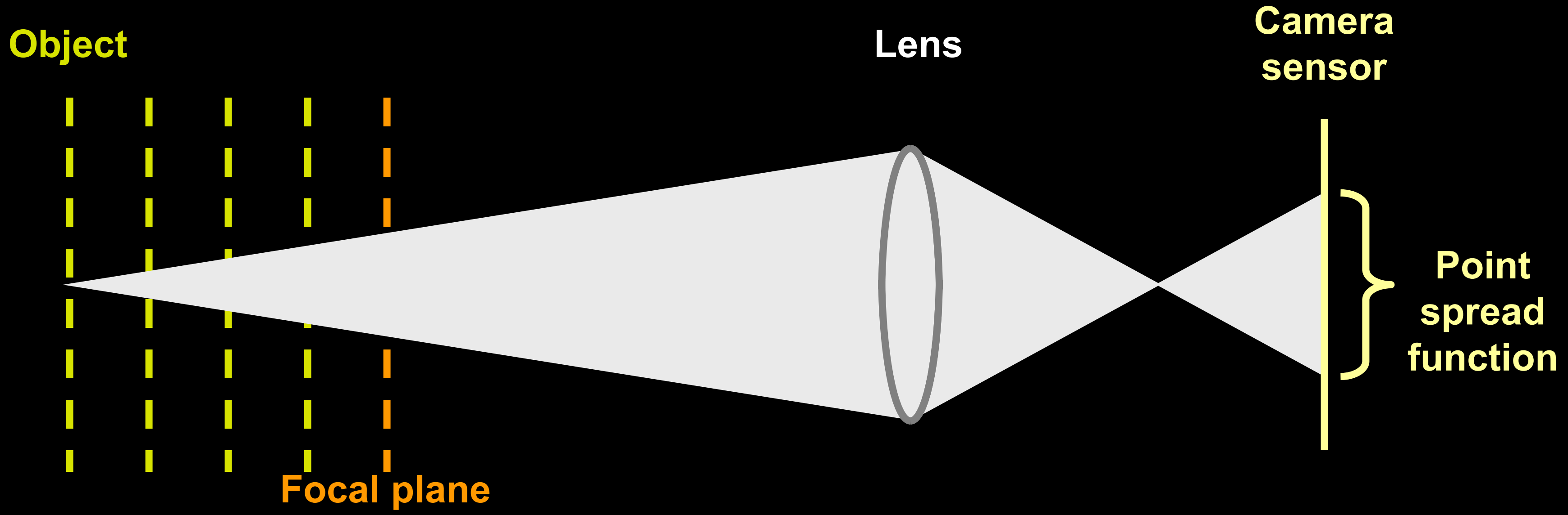


Image of a defocused point light source



Depth and defocus

Out of focus

In focus

ill posed

Depth from defocus:

Infer depth by analyzing local scale of defocus blur

Challenges

- Hard to discriminate a smooth scene from defocus blur

?

Out of focus

- Hard to undo defocus blur

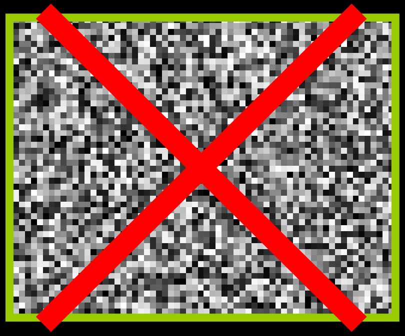
Input

Ringing with conventional deblurring algorithm

Key ideas

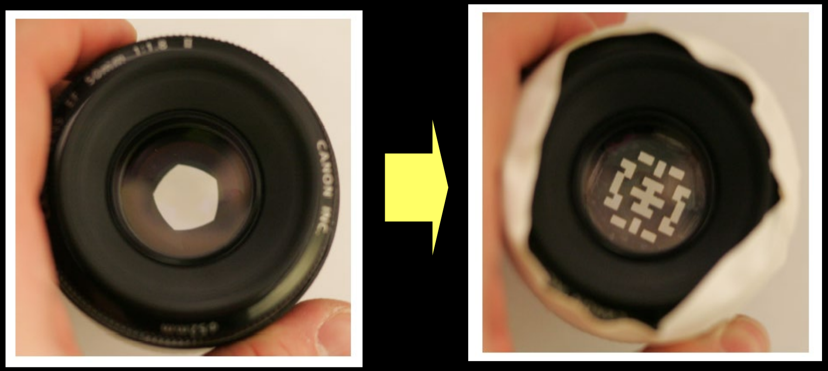
- Exploit prior on natural images
  - Improve deconvolution
  - Improve depth discrimination

Natural



Unnatural

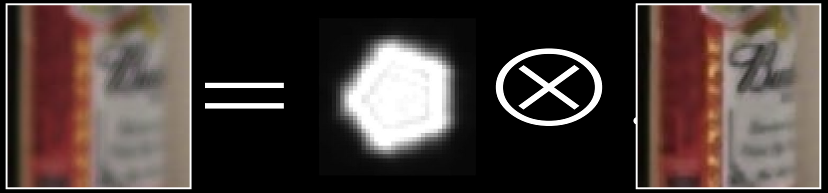
- Coded aperture (mask inside lens)
  - make defocus patterns different from natural images and easier to discriminate

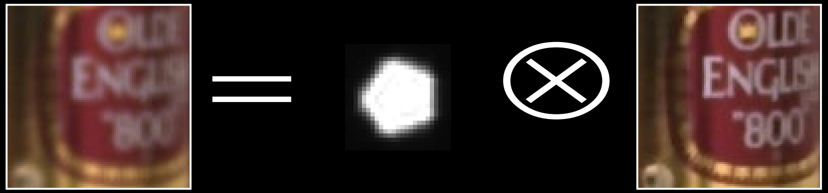


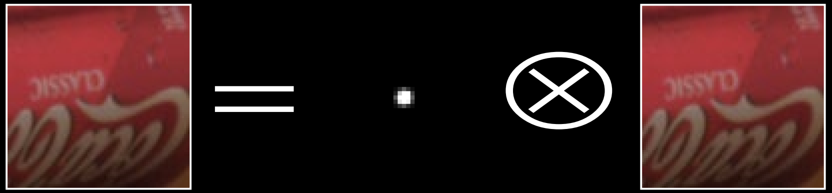
Defocus as local convolution

Input defocused image

$$\underset{\text{Local sub-window}}{y} = \underset{\text{Calibrated blur kernels at depth } k}{f_k} \otimes \underset{\text{Sharp sub-window}}{x}$$

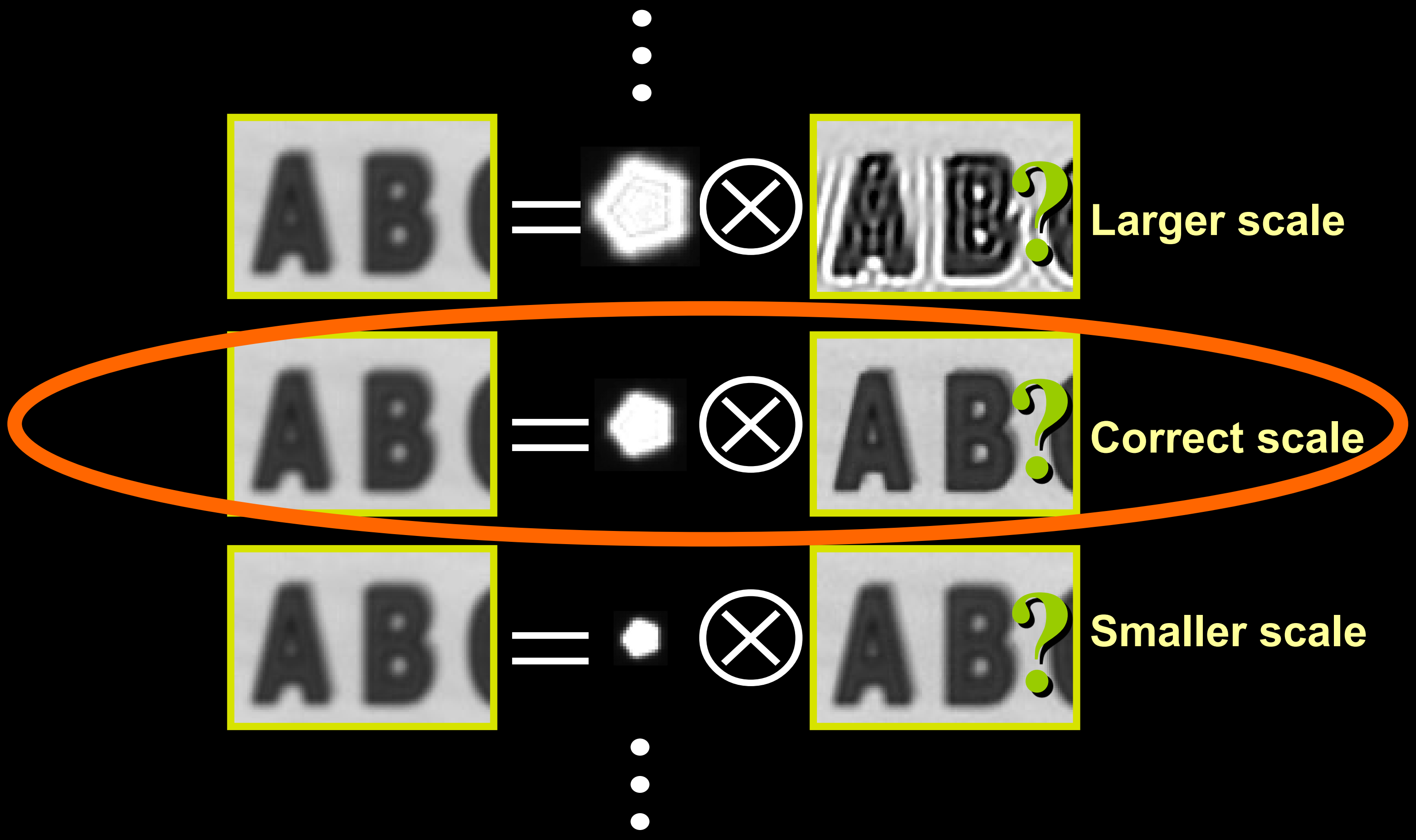
Depth k=1: 

Depth k=2: 

Depth k=3: 

Overview

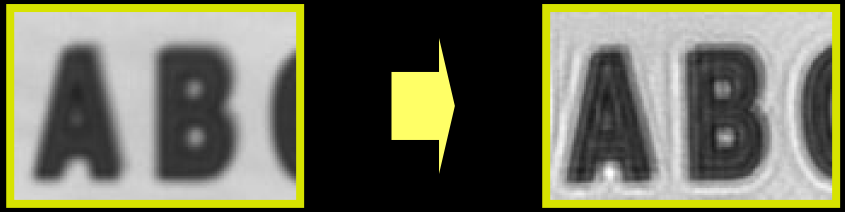
Try deconvolving local input windows with different scaled filters:



Somehow: select best scale.

Challenges

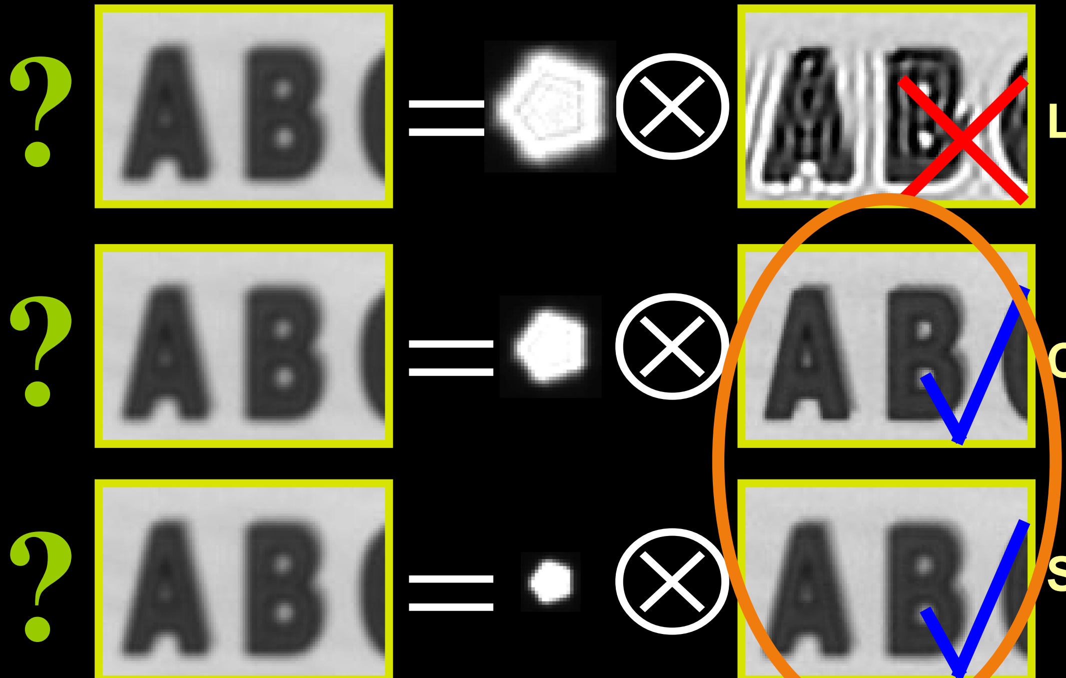
- Hard to deconvolve even when kernel is known



Input

Ringing with the traditional Richardson-Lucy deconvolution algorithm

- Hard to identify correct scale:



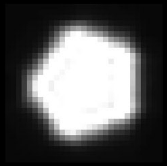
Larger scale

Correct scale

Smaller scale

Deconvolution is ill posed

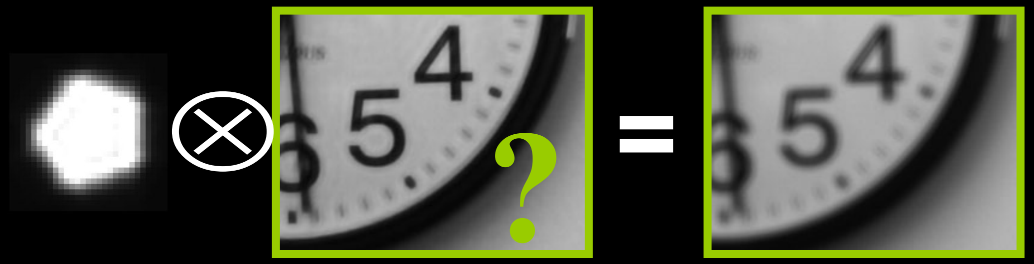
$$f \otimes x = y$$

 $\otimes$? = 

Deconvolution is ill posed

$$f \otimes x = y$$

Solution 1:



Solution 2:

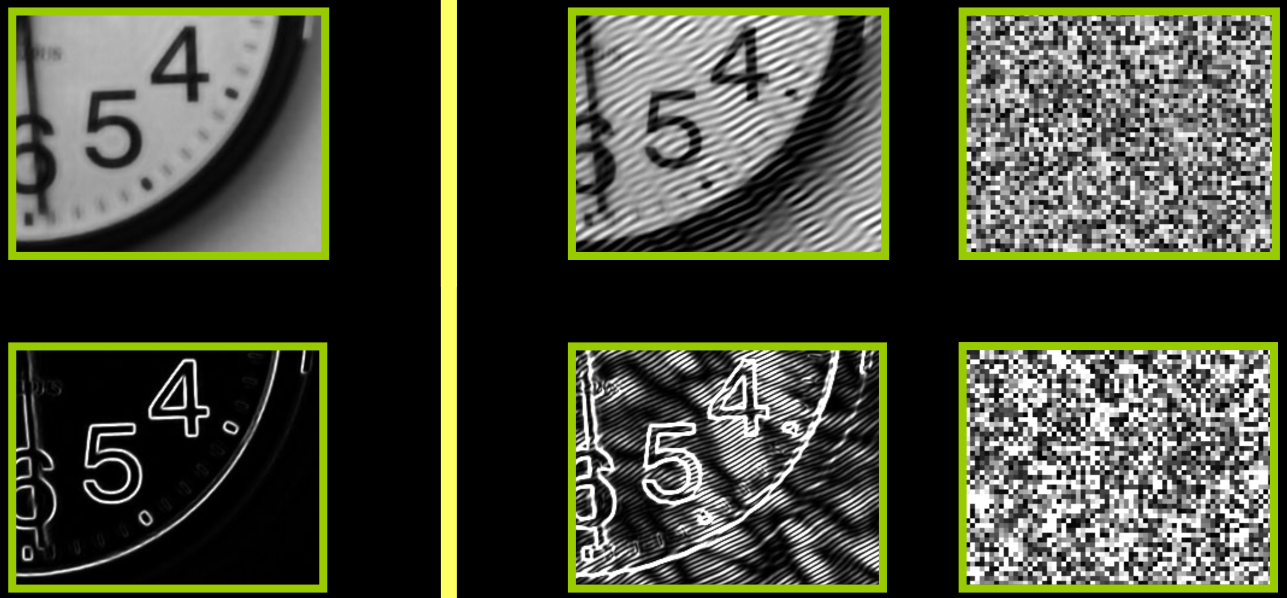
Idea 1: Natural images prior

What makes images special?

Natural | Unnatural

Image

gradient

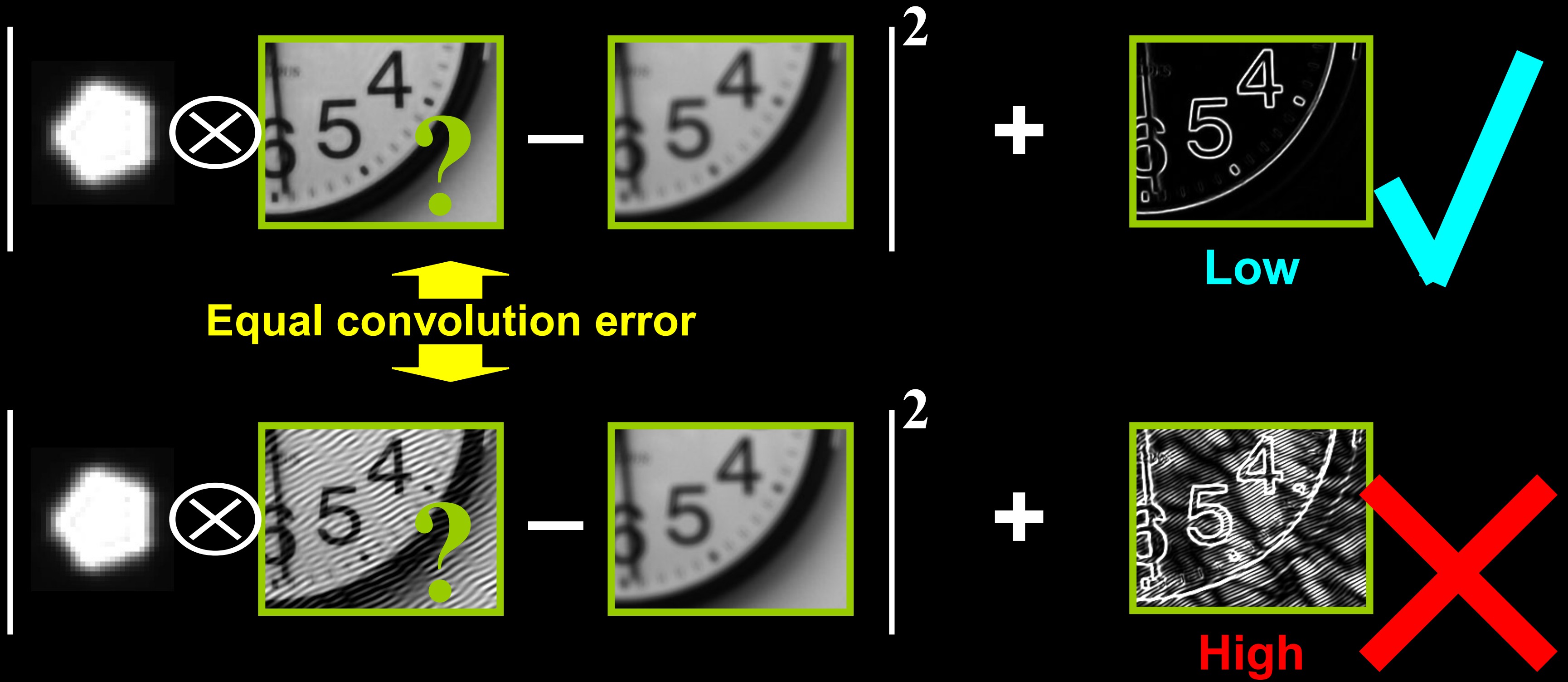


Natural images have sparse gradients

➡ put a penalty on gradients

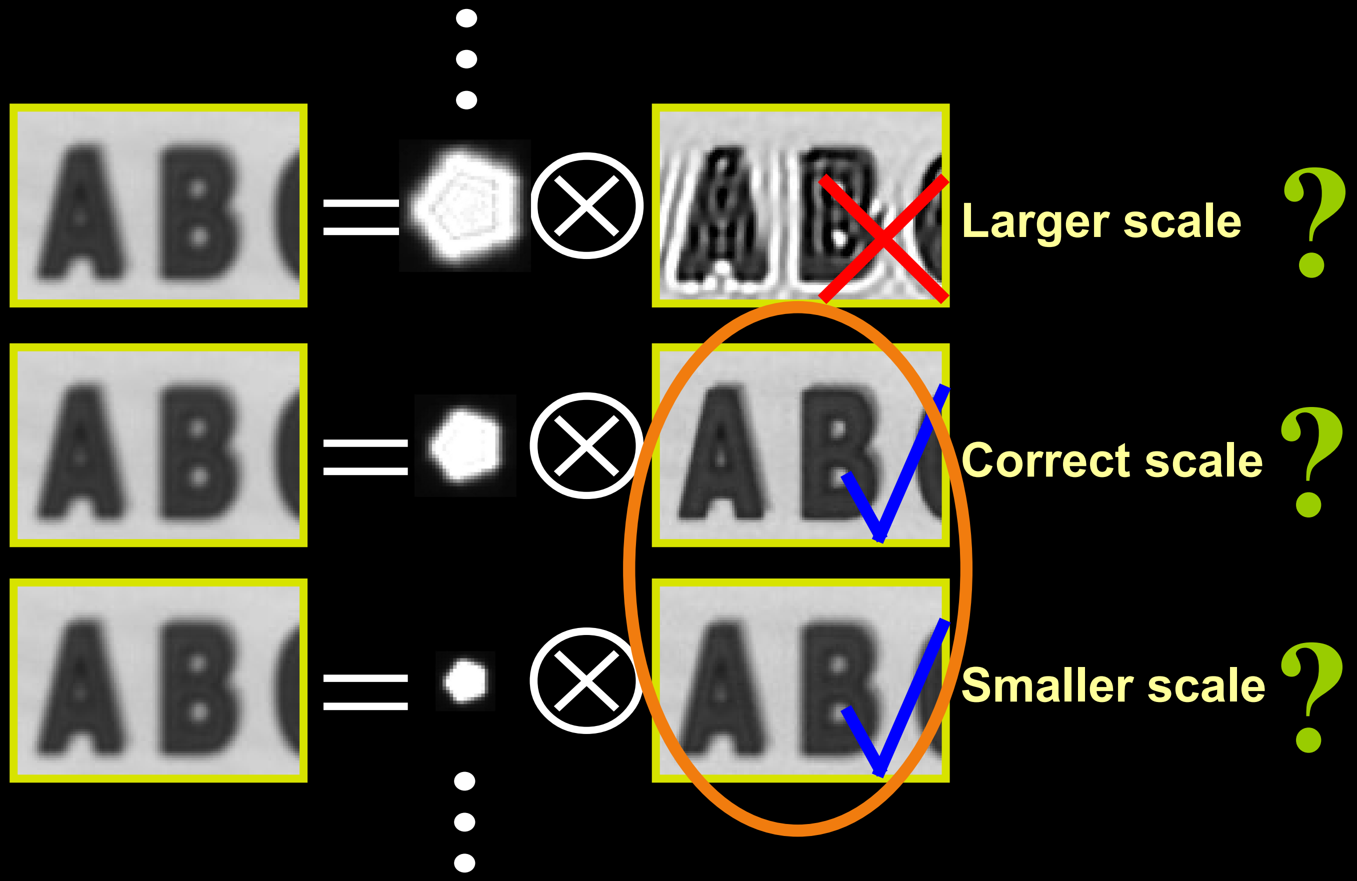
Deconvolution with prior

$$x = \arg\min \underbrace{|f \otimes x - y|^2}_{\text{Convolution error}} + \lambda \underbrace{\sum_i \rho(\nabla x_i)}_{\text{Derivatives prior}}$$



Recall: Overview

Try deconvolving local input windows with different scaled filters:



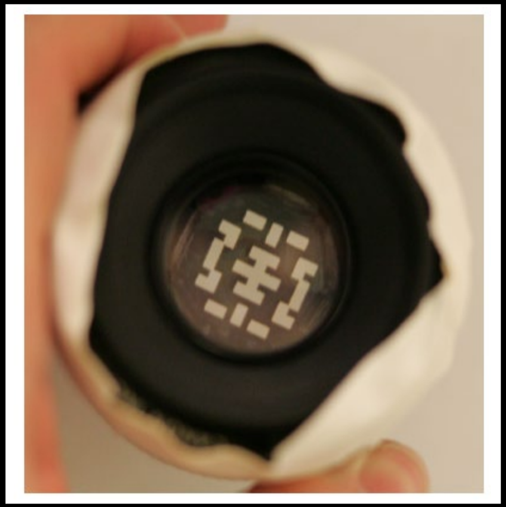
Somehow: select best scale.

Challenge: smaller scale not so different than correct

Idea 2: Coded Aperture

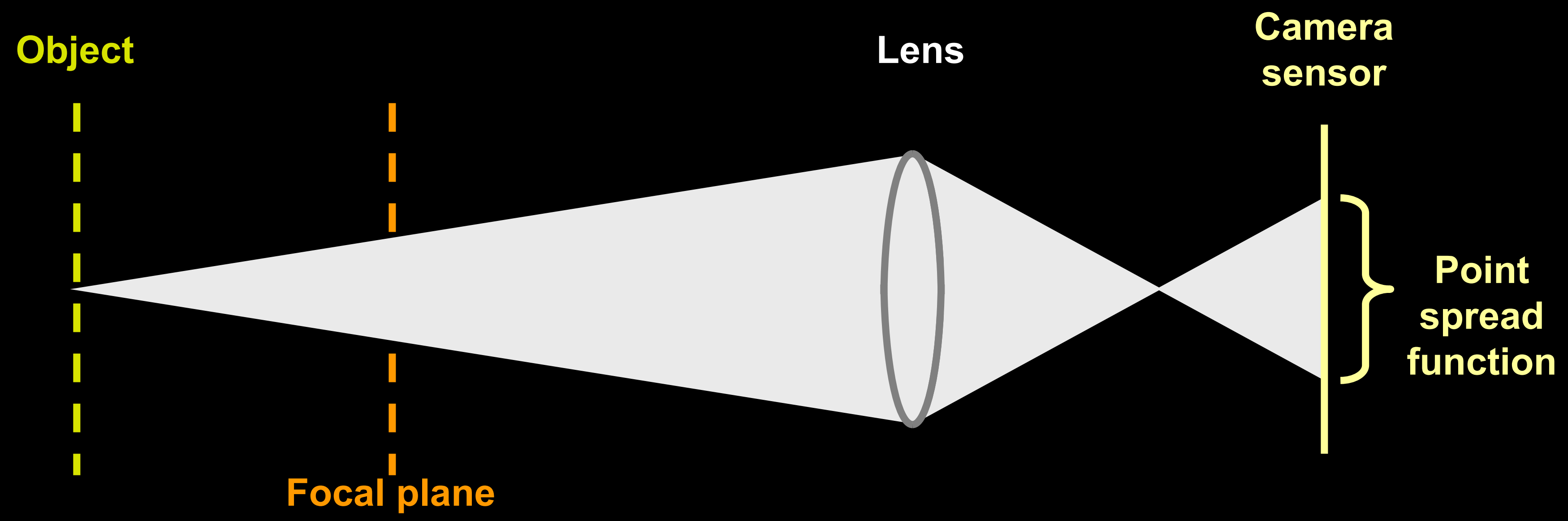
- Mask (code) in aperture plane
  - make defocus patterns different from natural images and easier to discriminate

Conventional aperture



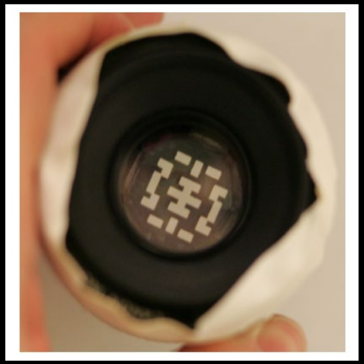
Our coded aperture

Solution: lens with occluder



Solution: lens with occluder

Aperture pattern



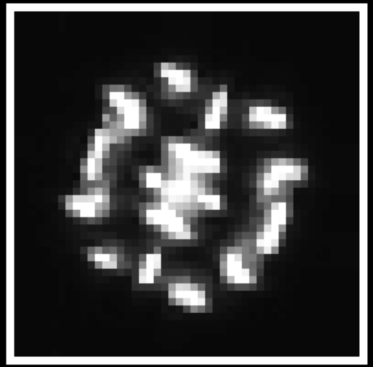
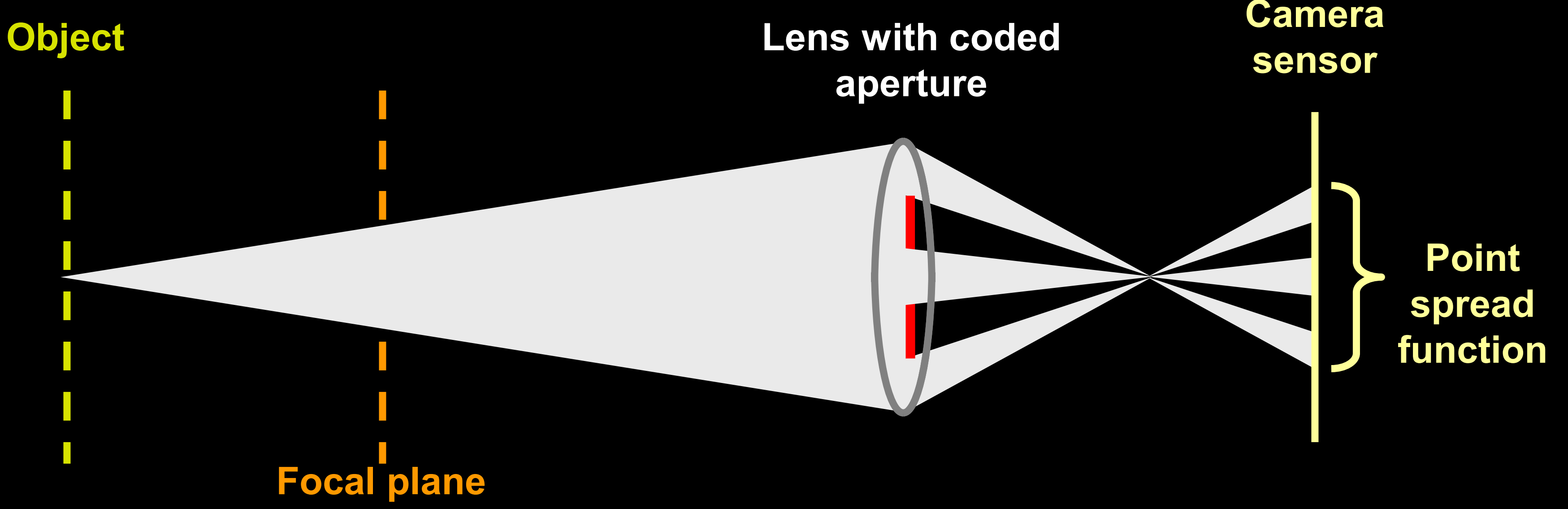
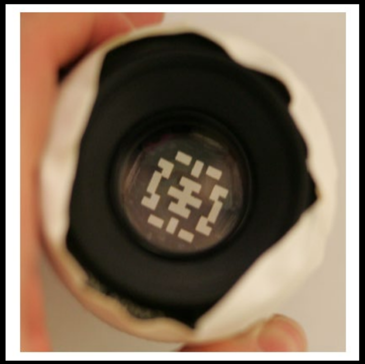


Image of a defocused point light source



Solution: lens with occluder

Aperture pattern



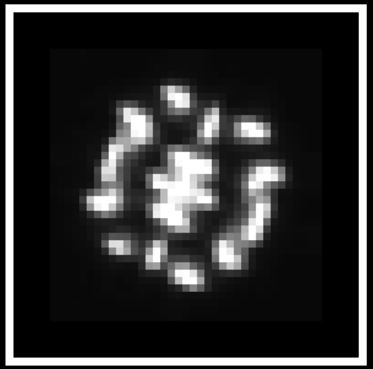
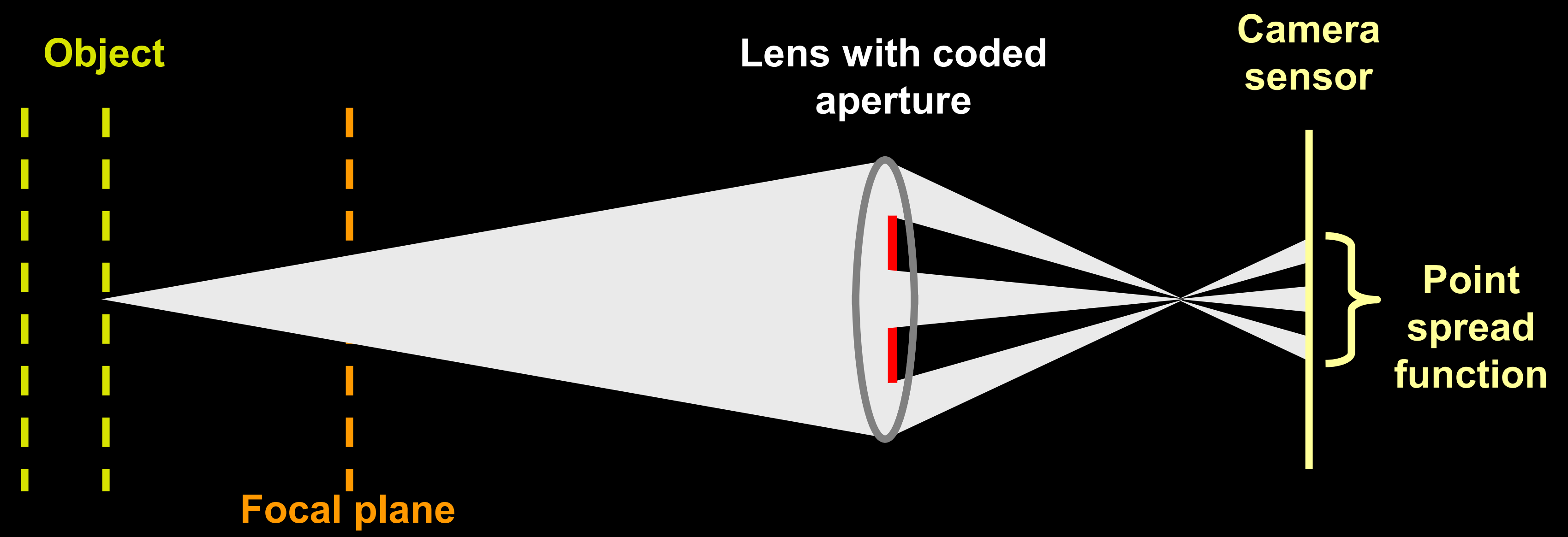
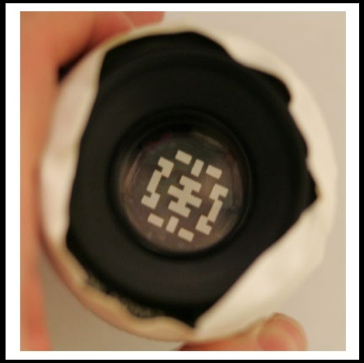


Image of a defocused point light source



Solution: lens with occluder

Aperture pattern



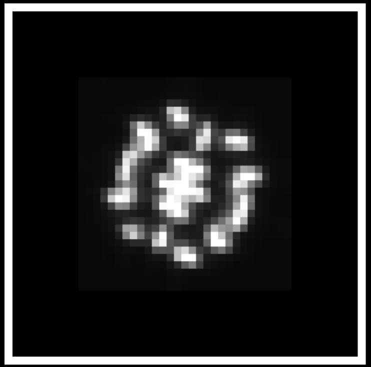
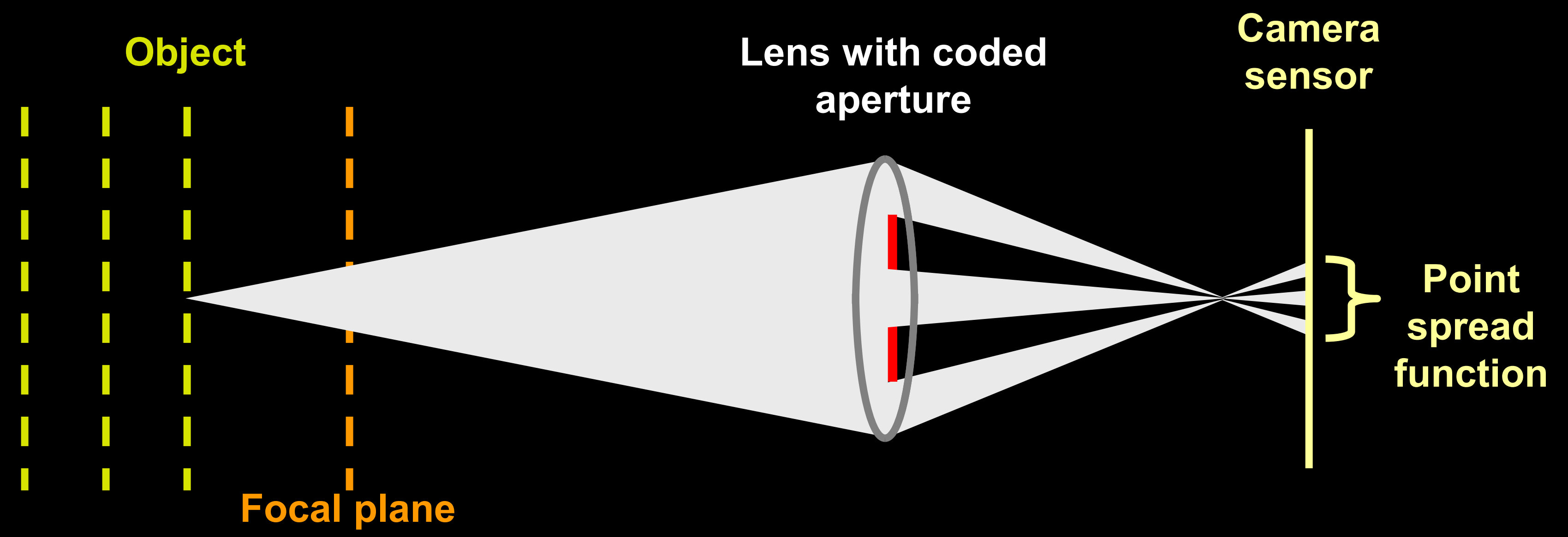
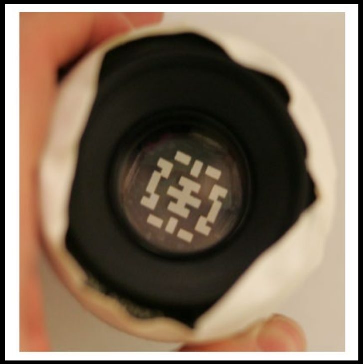


Image of a defocused point light source



Solution: lens with occluder

Aperture pattern



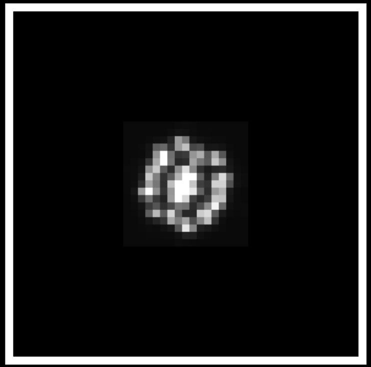
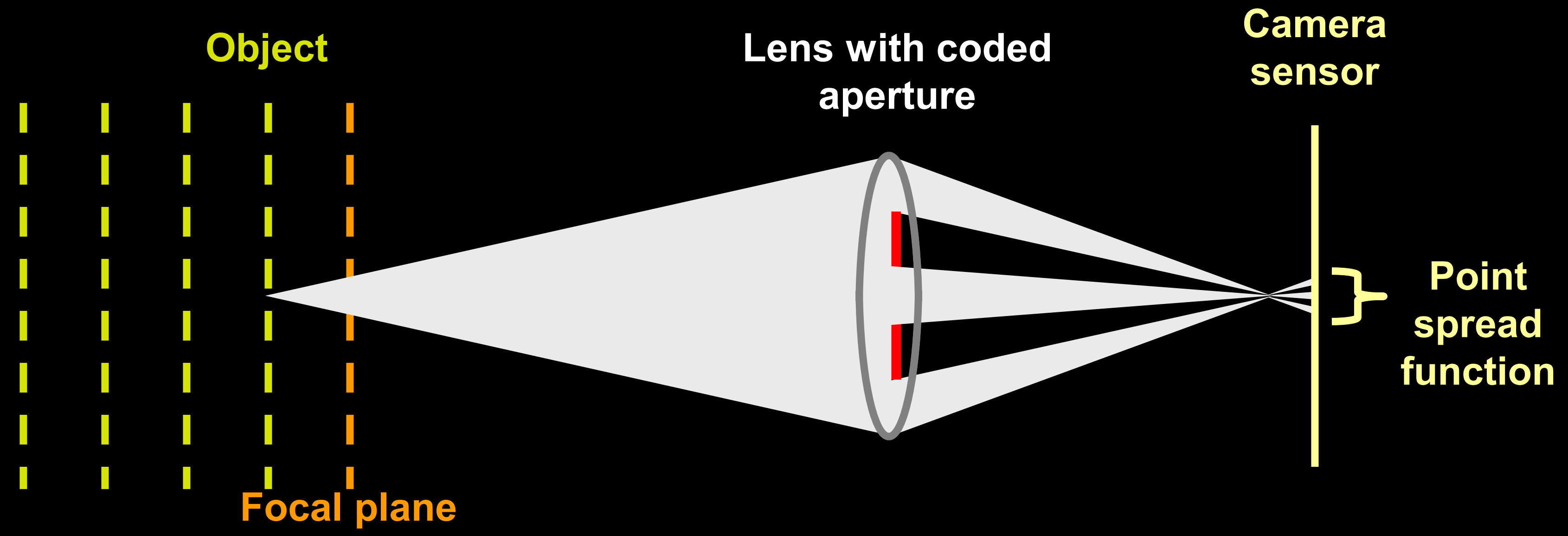


Image of a defocused point light source



Solution: lens with occluder

Aperture pattern

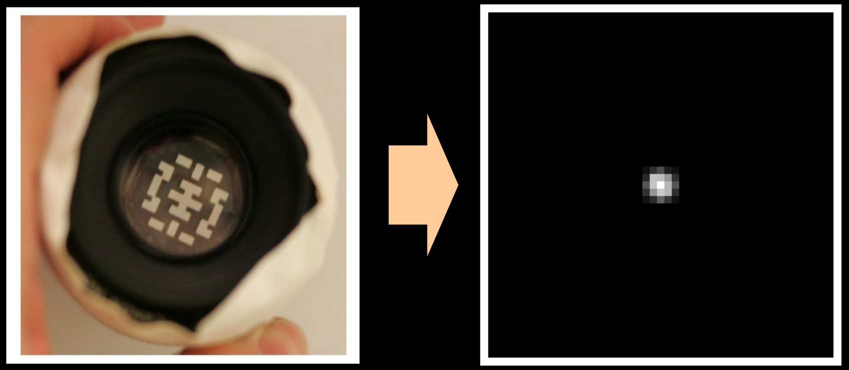
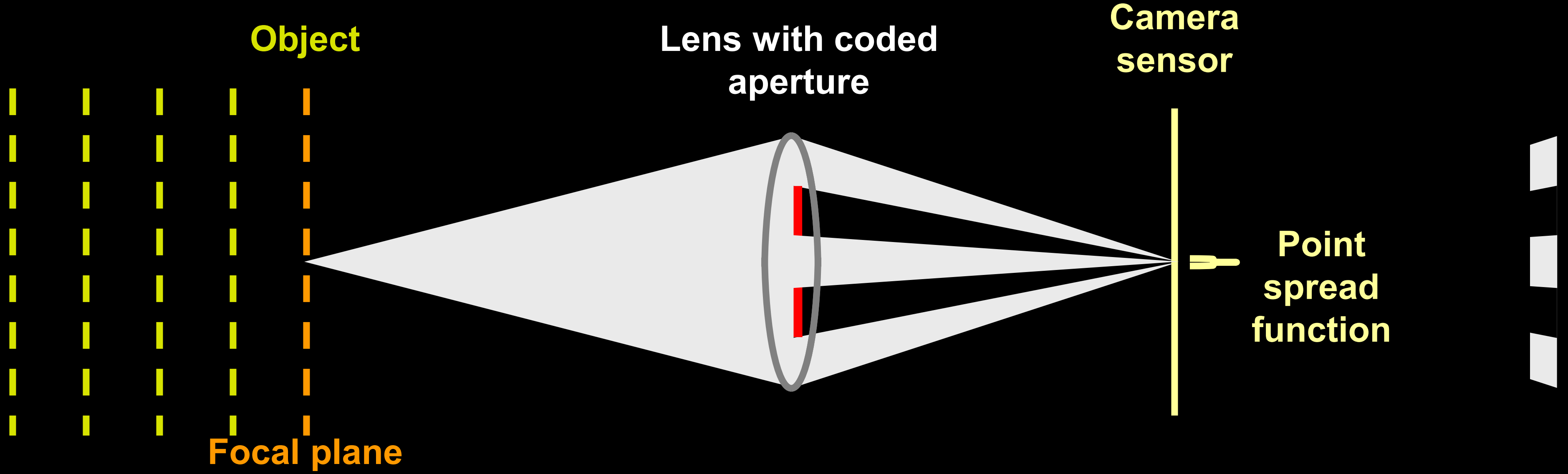
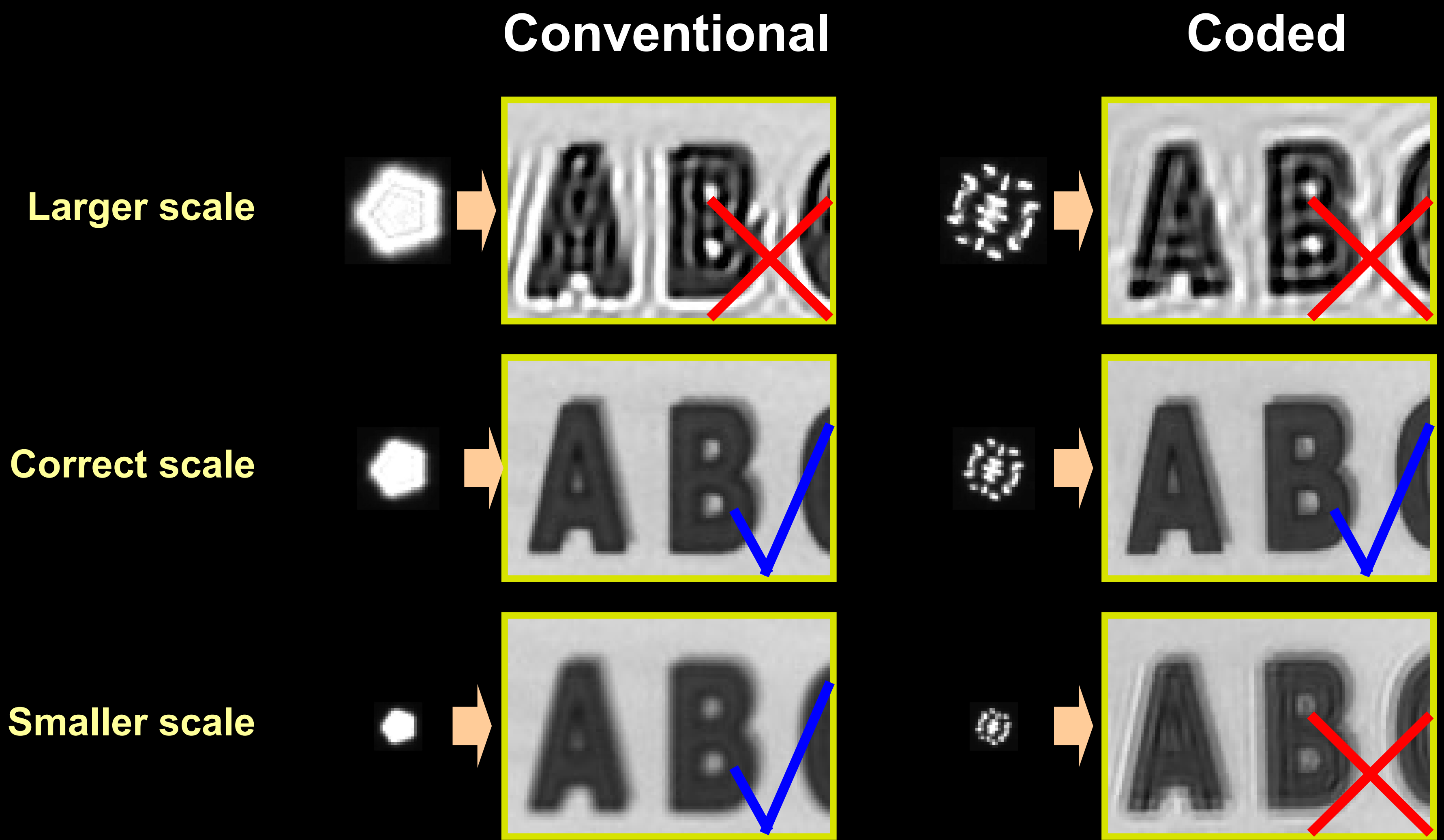


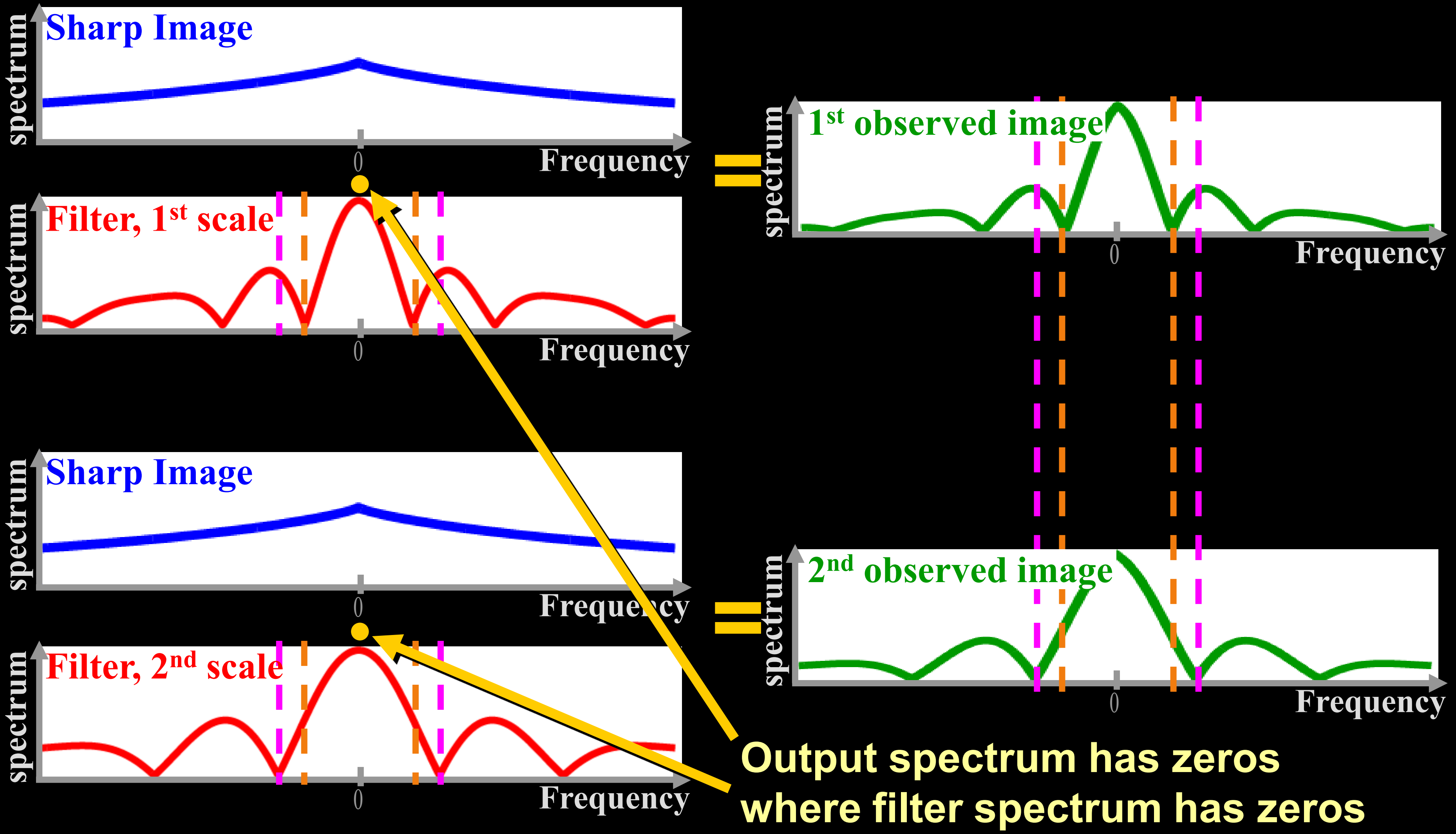
Image of a defocused point light source



Coded aperture reduces uncertainty in scale identification

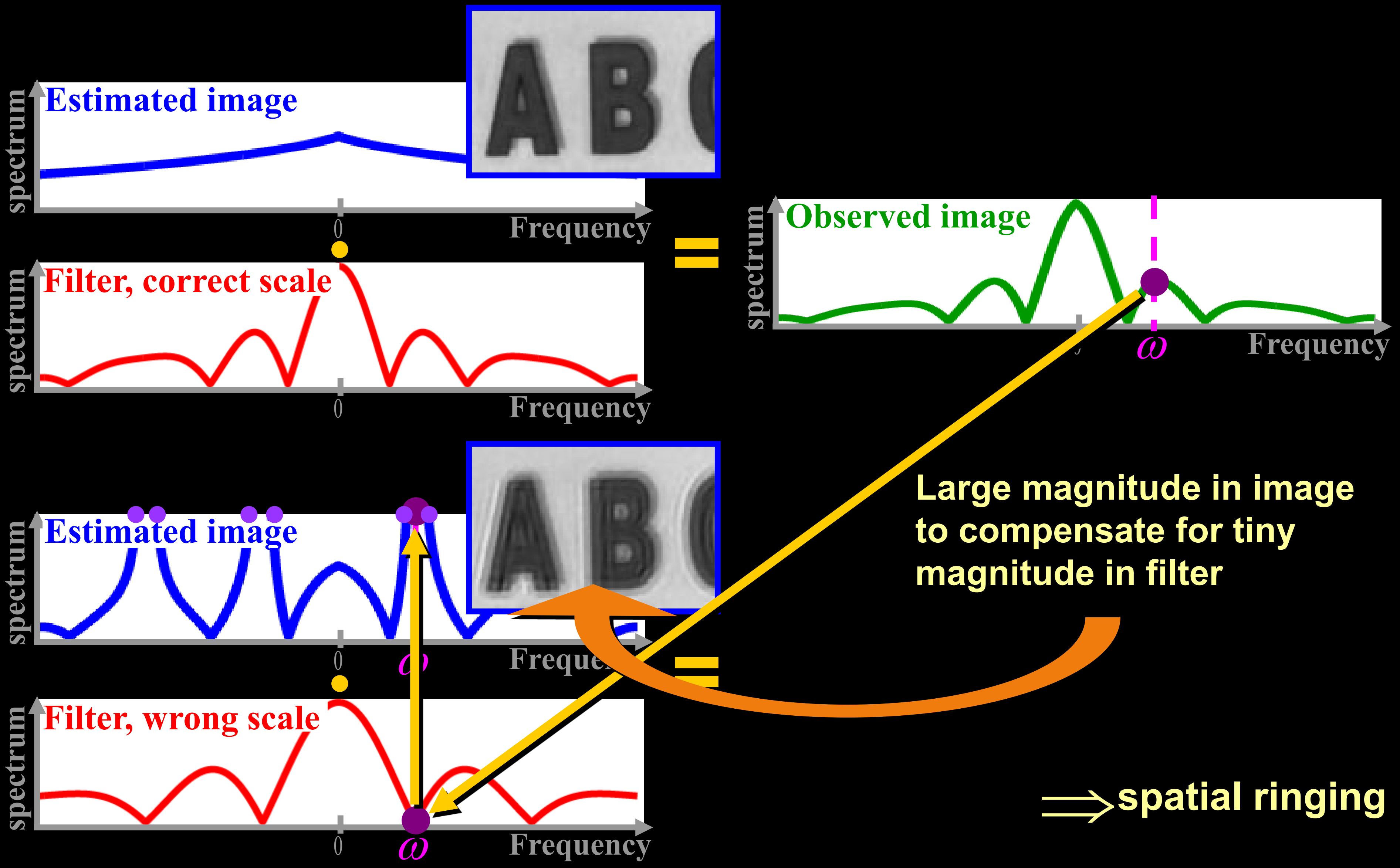


Convolution- frequency domain representation



Spatial convolution ⟺ frequency multiplication

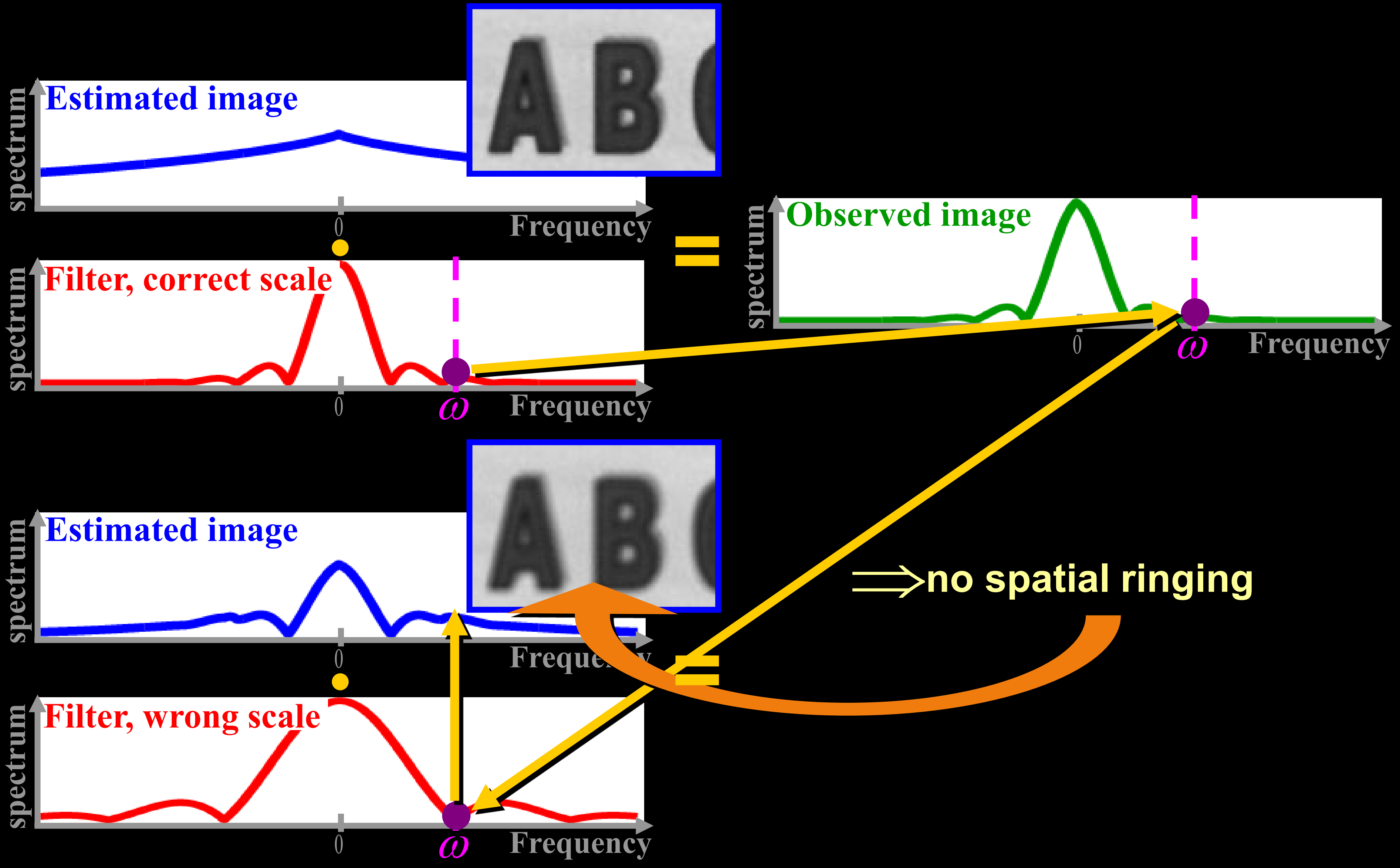
Coded aperture: Scale estimation and division by zero



Large magnitude in image to compensate for tiny magnitude in filter

$\Longrightarrow$ spatial ringing

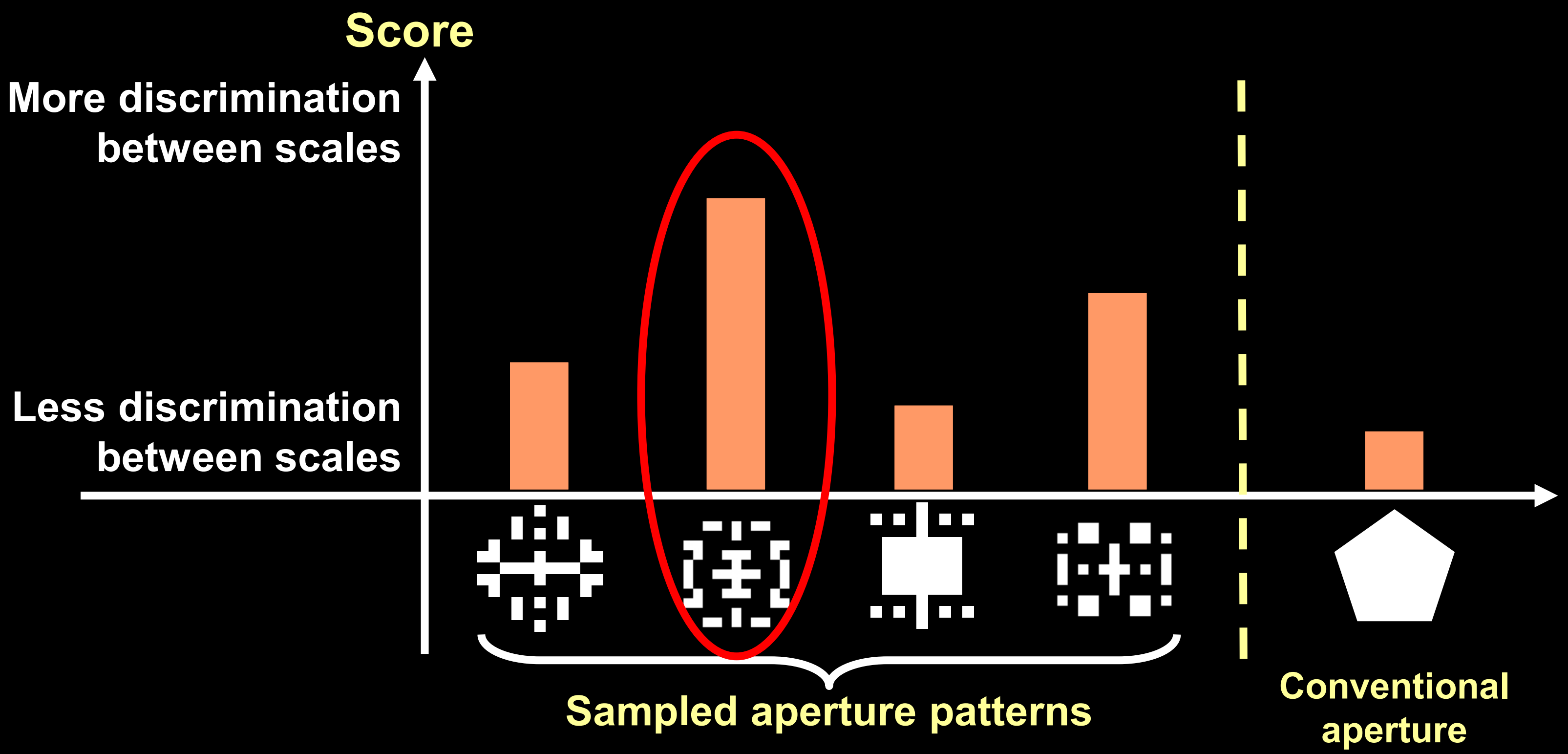
Division by zero with a conventional aperture?



Filter Design

Analytically search for a pattern maximizing discrimination between images at different defocus scales (*KL-divergence)*

Account for image prior and physical constraints

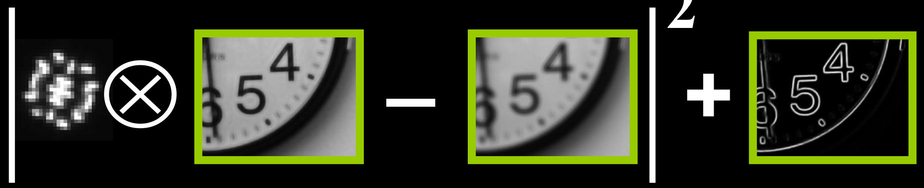


Depth results

Regularizing depth estimation

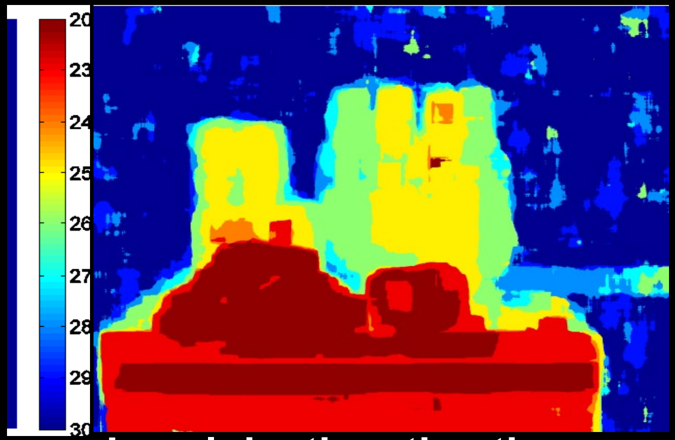
Try deblurring with 10 different aperture scales

$$x = \arg\min \underbrace{|f \otimes x - y|^2}_{\text{Convolution error}} \; + \underbrace{\lambda \sum_i \rho(\nabla x_i)}_{\text{Derivatives prior}}$$



Keep minimal error scale in each local window + regularization

Input

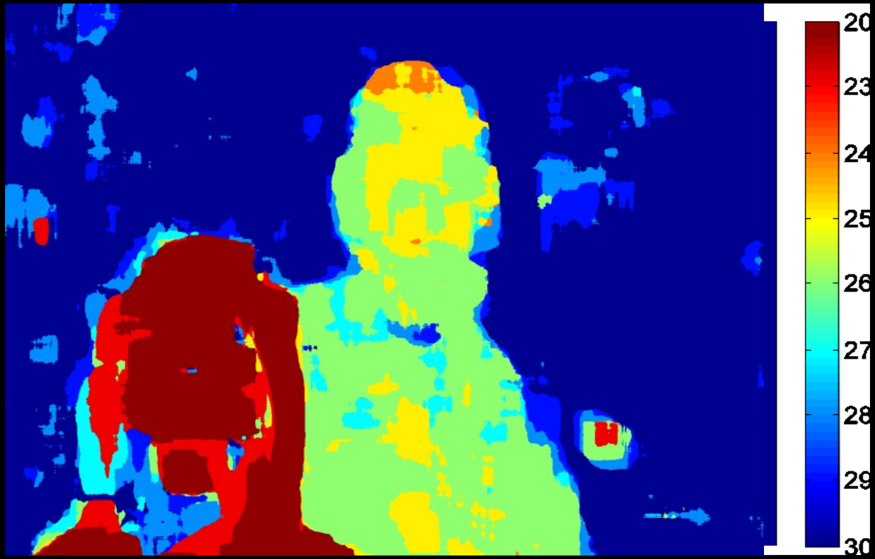


Local depth estimation

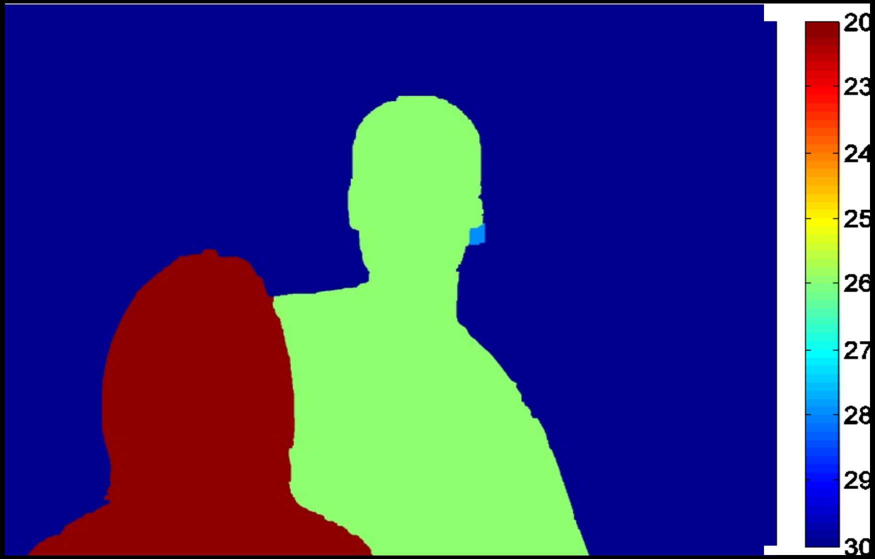
Regularized depth

Regularizing depth estimation

Input



Local depth estimation



Regularized depth

All focused results

Input

All-focused (deconvolved)

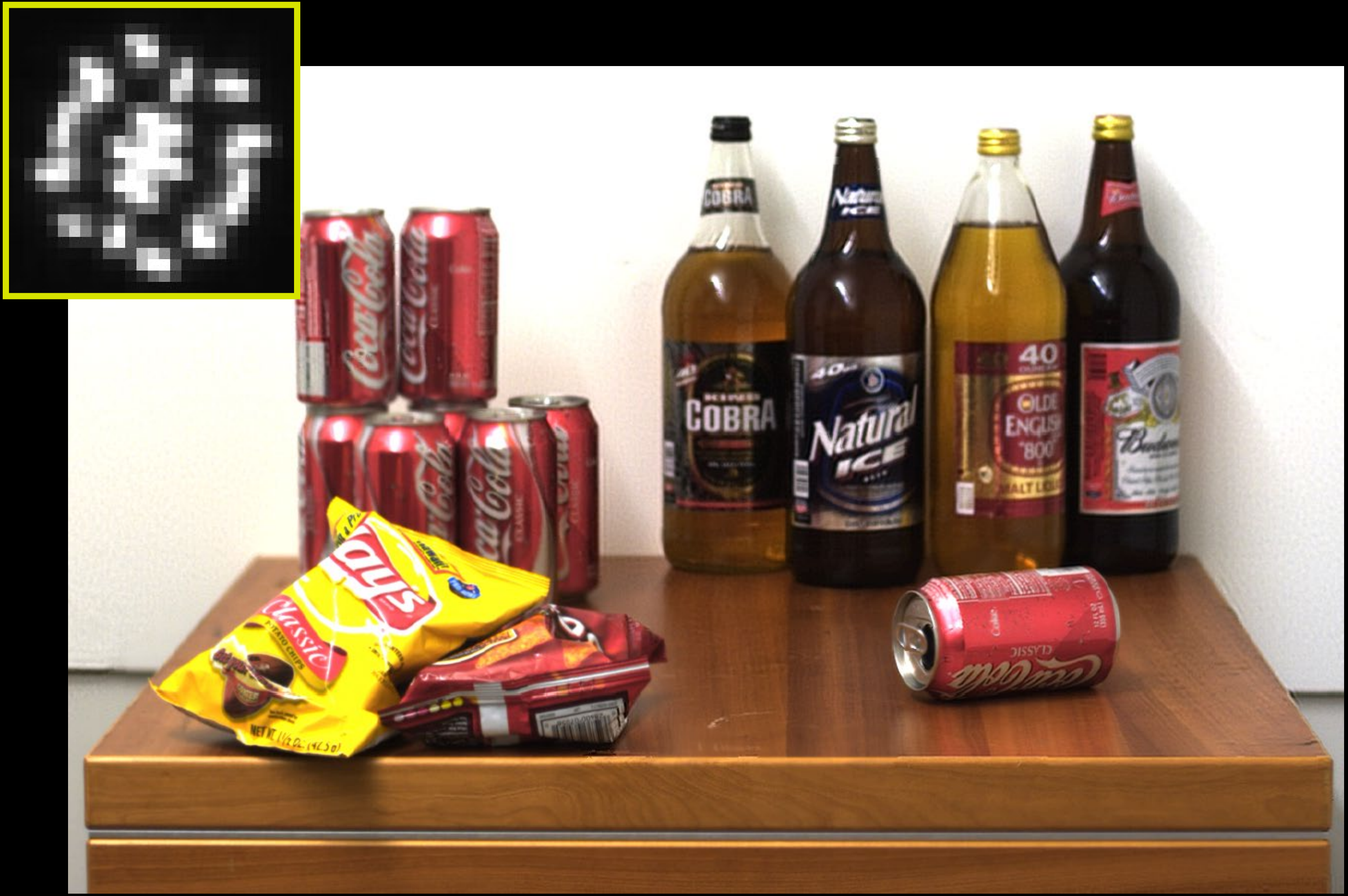
Close-up

Original image

All-focus image

Comparison- conventional aperture result

Comparison- conventional aperture result




Input


All-focused
(deconvolved)

Close-up

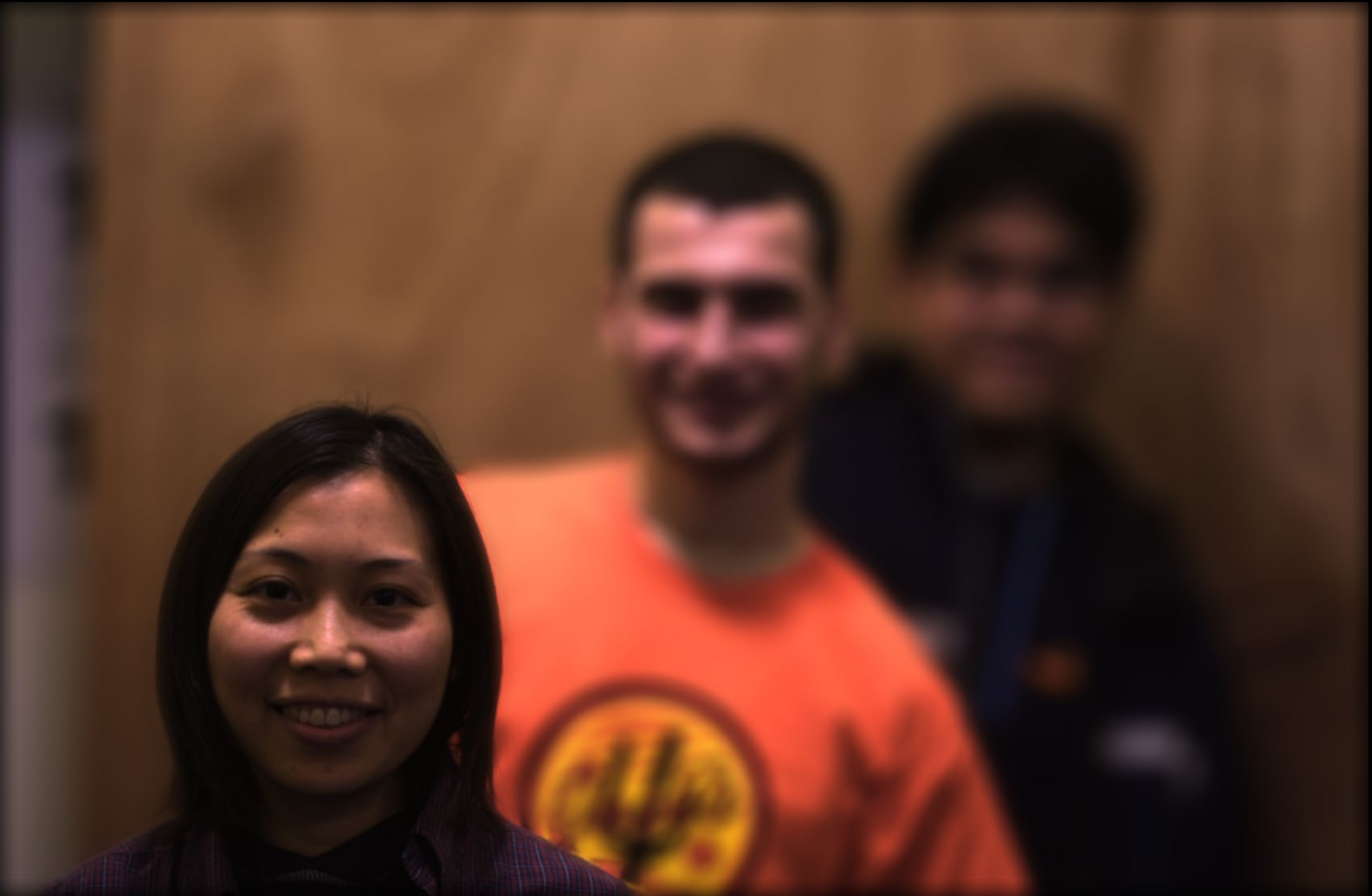
Original image

All-focus image

Naïve sharpening

Application: Digital refocusing from a single image

Application: Digital refocusing from a single image

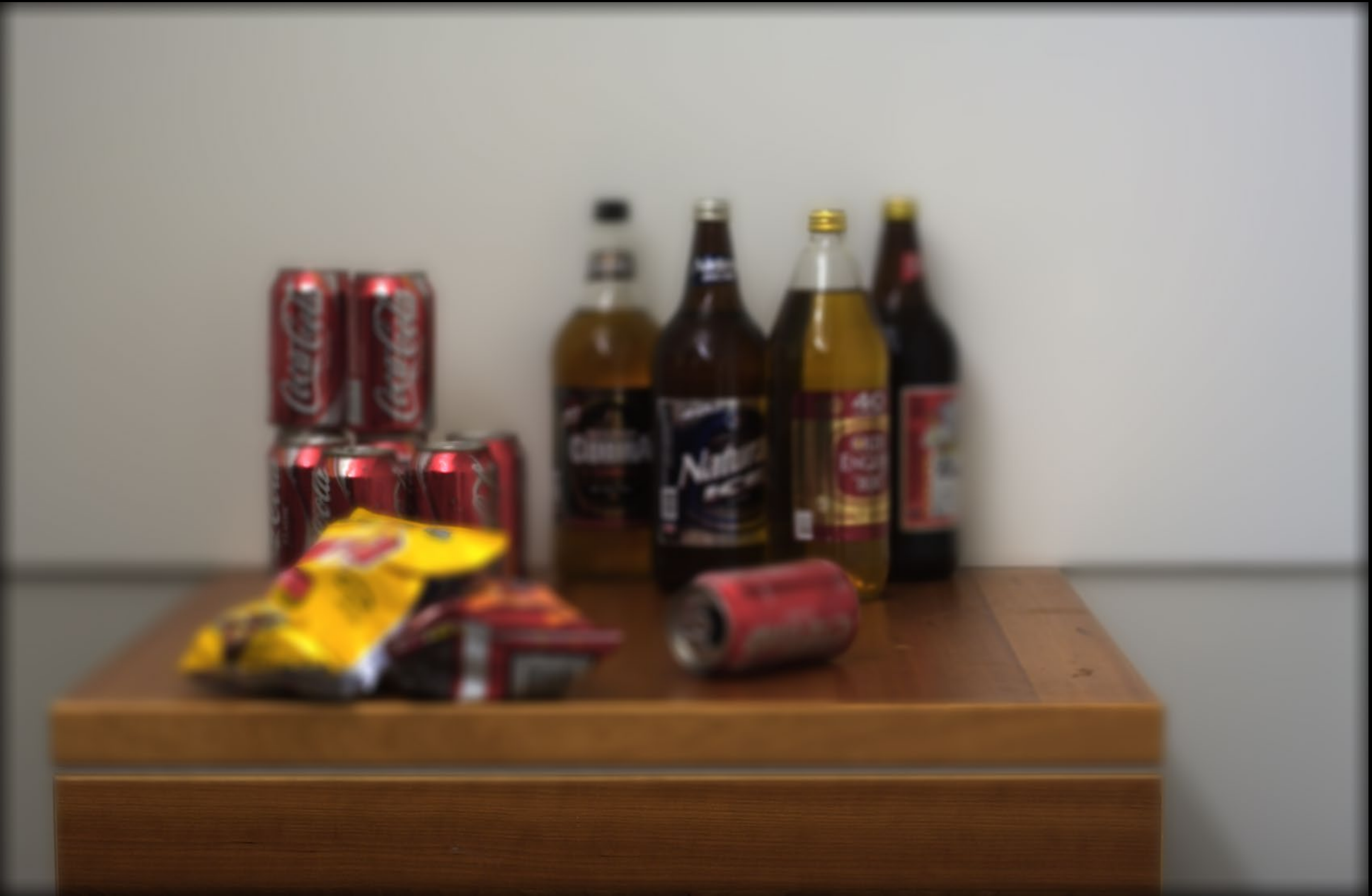


Application: Digital refocusing from a single image

Application: Digital refocusing from a single image

Application: Digital refocusing from a single image

Application: Digital refocusing from a single image



Application: Digital refocusing from a single image

Coded aperture: pros and cons

+ Image AND depth at a single shot
+ No loss of image resolution
+ Simple modification to lens
- Depth is coarse

  unable to get depth at untextured areas,
  might need manual corrections.

+ But depth is a pure bonus
- Lose some light
+ But deconvolution increases depth of field

50mm f/1.8: $79.95
Cardboard: $1
Tape: $1
Depth acquisition: priceless

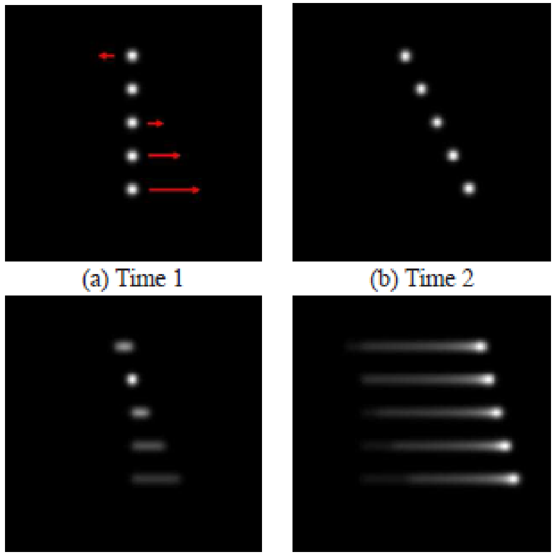
CodedAperture

Some more quick examples

Motion-Invariant Photography

Anat Levin Peter Sand Taeg Sang Cho Frédo Durand William T. Freeman
Massachusetts Institute of Technology, Computer Science and Artificial Intelligence Laboratory

- Quickly move camera in a parabola when taking a picture
- A motion at any speed in the direction of the parabola will give the same blur kernel



(a) Time 1 (b) Time 2

(c) Static camera (d) Parabolic camera

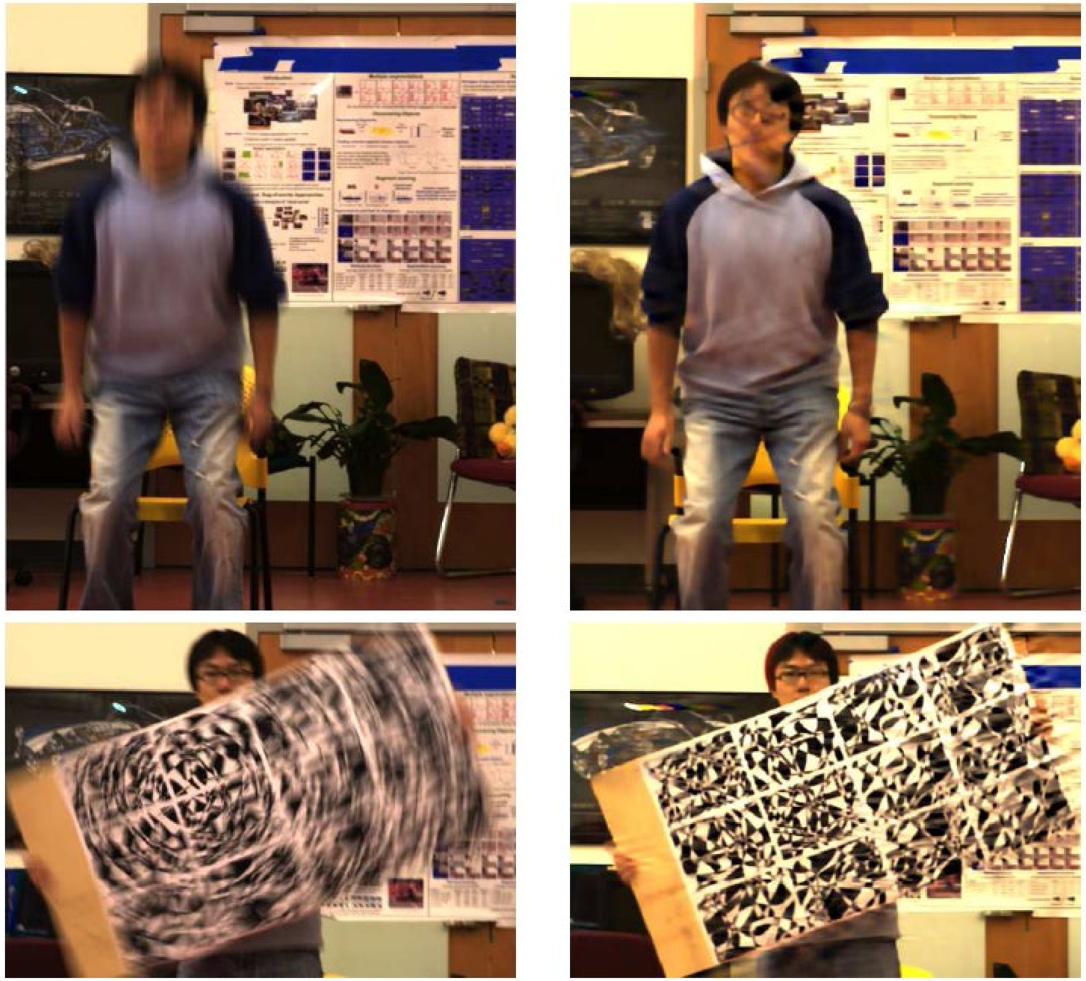
Results

Static Camera

Parabolic Camera

Results

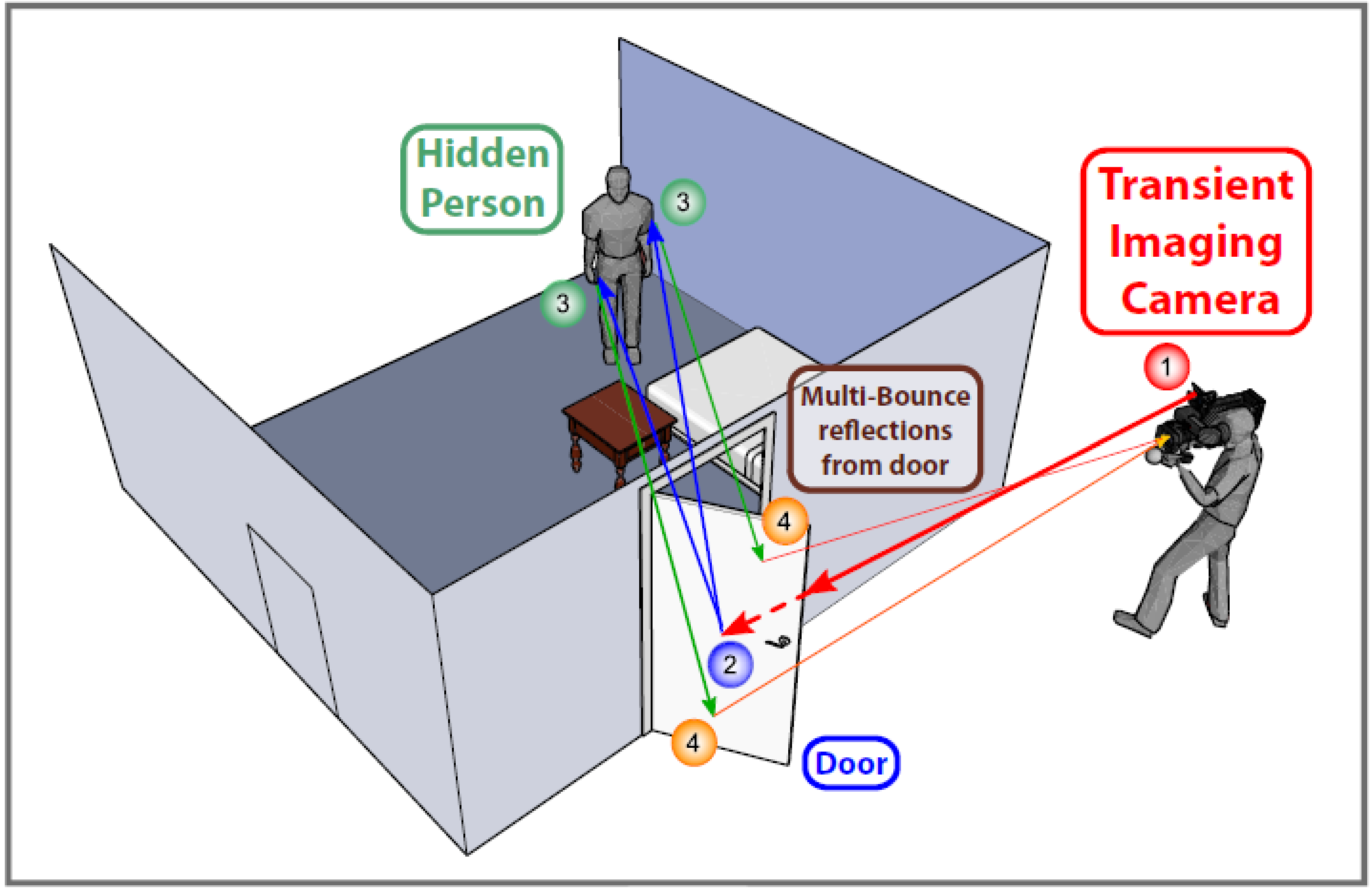
Static Camera Parabolic Camera



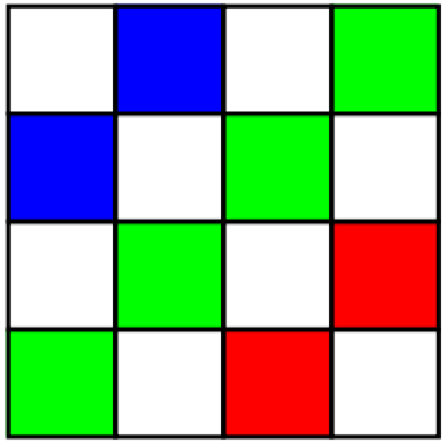
Motion in wrong direction

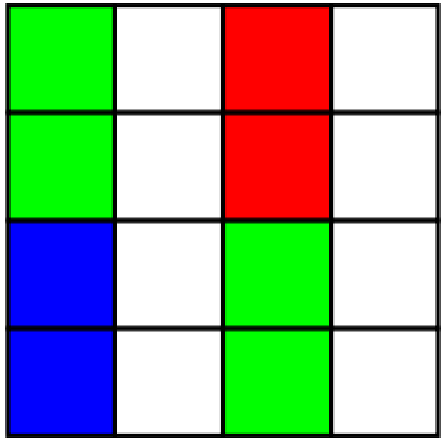
Looking Around the Corner using Transient Imaging

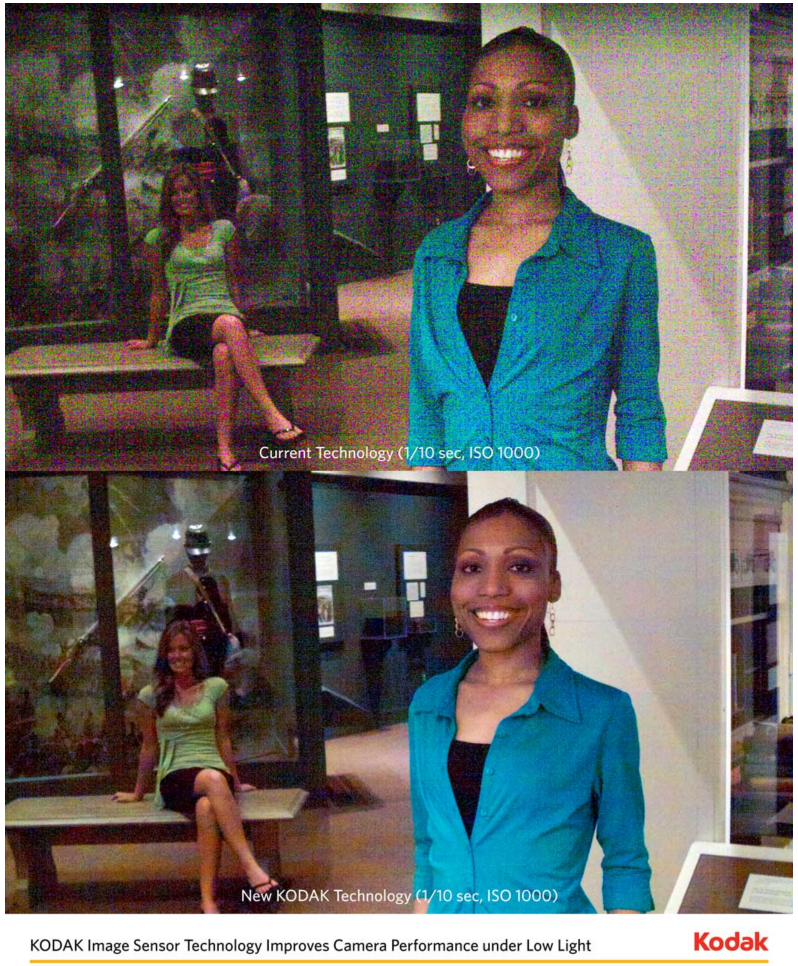
Ahmed Kirmani *[1], Tyler Hutchison[1], James Davis †[2], and Ramesh Raskar‡[1]
[1]MIT Media Laboratory [2] UC Santa Cruz



RGBW Sensors







- 2007: Kodak 'Panchromatic' Pixels
- Outperforms Bayer Grid
  - 2X-4X sensitivity (W: no filter loss)
  - May improve dynamic range (W >> RGB sensitivity)

http://www.dpreview.com/news/2007/6/14/kodakhighsens

Computational Approaches to Display

- 3D TV without glasses
  - 20”, $2900, available in Japan (2010)
  - You see different images from different angles

Toshiba

http://news.cnet.com/8301-13506_3-20018421-17.html
Newer version: http://www.pcmag.com/article2/0,2817,2392380,00.asp
http://reviews.cnet.com/3dtv-buying-guide/

Recap of questions

- How can we represent all of the information contained in light?
- What are the fundamental limitations of cameras?
- What sacrifices have we made in conventional cameras? For what benefits?
- How else can we design cameras for better focus, deblurring, multiple views, depth, etc.?

Next class

- Exam review