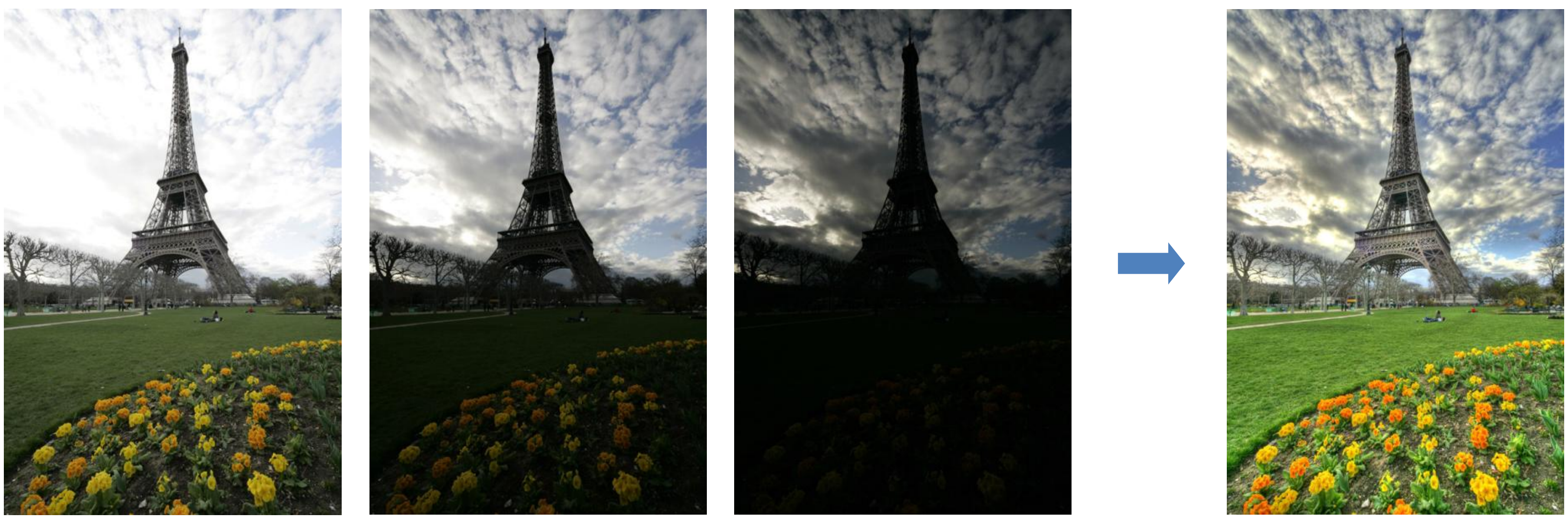
CS6670: Computer Vision

Noah Snavely

Lecture 22: Computational photography

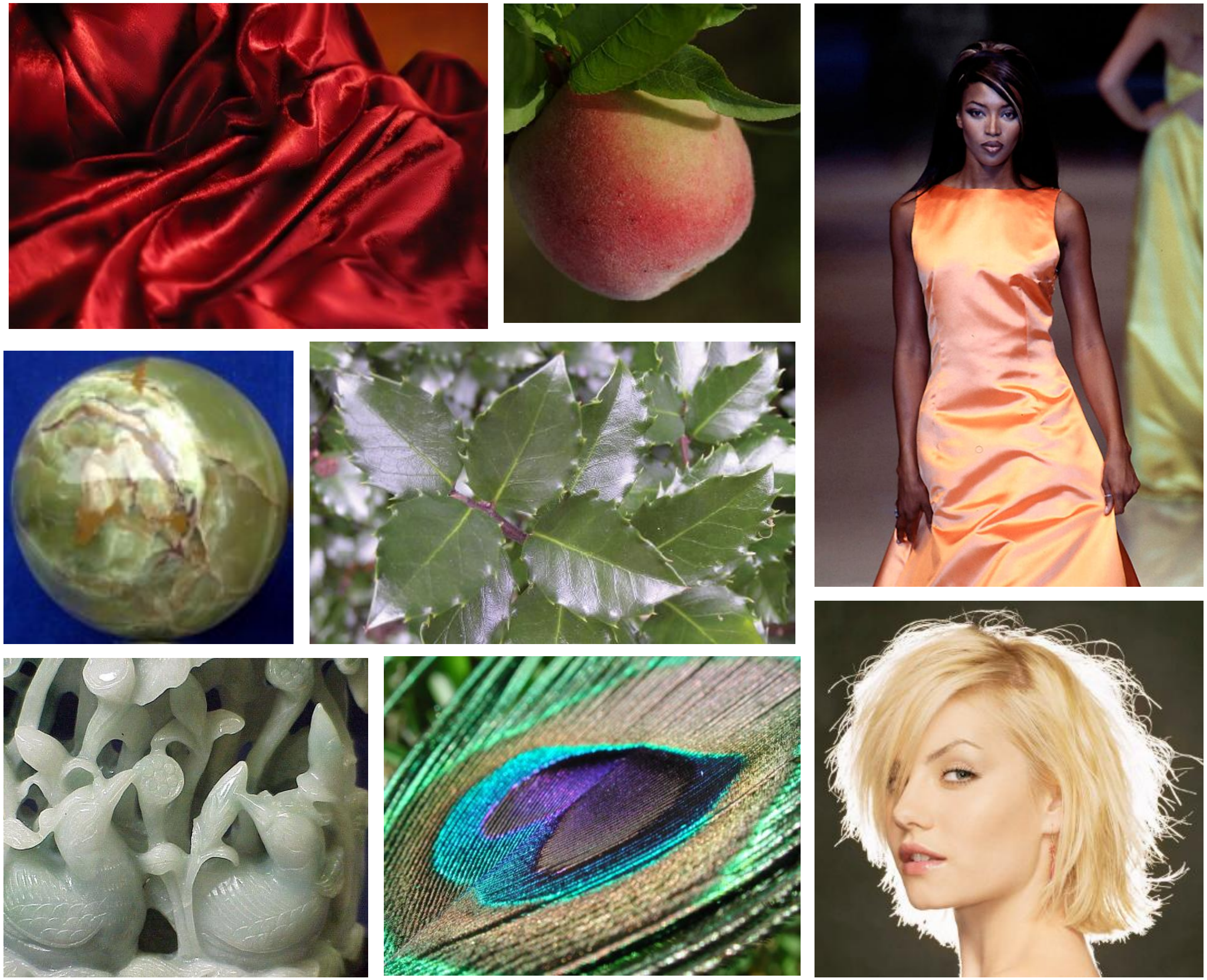


photomatix.com

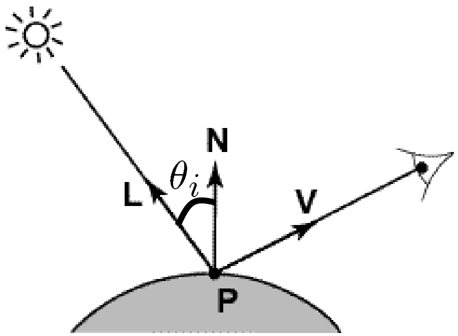
Announcements

- Final project midterm reports due on Tuesday to CMS by 11:59pm

BRDF’s can be incredibly complicated…



Shape from shading



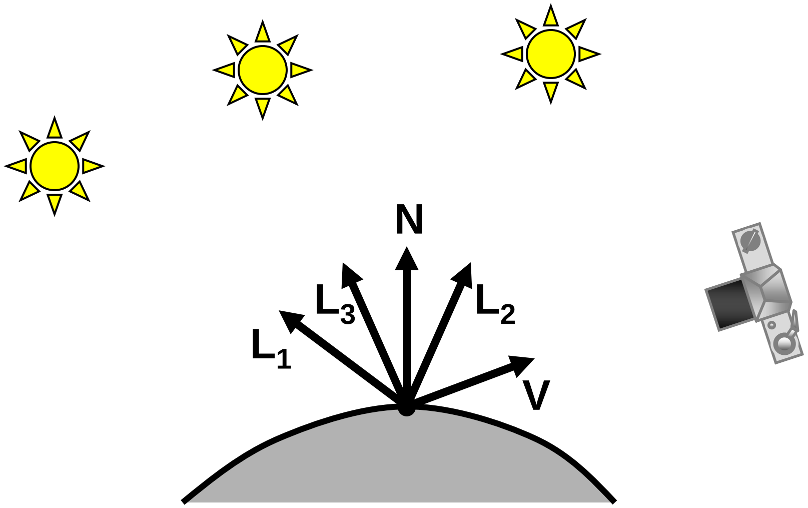
Suppose $k_d = 1$

$$\begin{aligned} I &= k_d \mathbf{N} \cdot \mathbf{L} \\ &= \mathbf{N} \cdot \mathbf{L} \\ &= cos\ \theta_i \end{aligned}$$

You can directly measure angle between normal and light source

- Not quite enough information to compute surface shape
- But can be if you add some additional info, for example
  - assume a few of the normals are known (e.g., along silhouette)
  - constraints on neighboring normals—“integrability”
  - smoothness
- Hard to get it to work well in practice
  - plus, how many real objects have constant albedo?

Photometric stereo



$$I_1 = k_d \mathbf{N} \cdot \mathbf{L_1}$$
$$I_2 = k_d \mathbf{N} \cdot \mathbf{L_2}$$
$$I_3 = k_d \mathbf{N} \cdot \mathbf{L_3}$$

Can write this as a matrix equation:

$$\begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = k_d \begin{bmatrix} \mathbf{L_1}^T \\ \mathbf{L_2}^T \\ \mathbf{L_3}^T \end{bmatrix} \mathbf{N}$$

Solving the equations

$$\underbrace{\begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix}}_{\substack{\mathbf{I} \\ 3\times 1}} = \underbrace{\begin{bmatrix} \mathbf{L_1}^T \\ \mathbf{L_2}^T \\ \mathbf{L_3}^T \end{bmatrix}}_{\substack{\mathbf{L} \\ 3\times 3}} \underbrace{k_d \mathbf{N}}_{\substack{\mathbf{G} \\ 3\times 1}}$$

$$\mathbf{G} = \mathbf{L}^{-1}\mathbf{I}$$

$$k_d = \|\mathbf{G}\|$$

$$\mathbf{N} = \frac{1}{k_d}\mathbf{G}$$

More than three lights

Get better results by using more lights

$$\begin{bmatrix} I_1 \\ \vdots \\ I_n \end{bmatrix} = \begin{bmatrix} \mathbf{L_1} \\ \vdots \\ \mathbf{L_n} \end{bmatrix} k_d \mathbf{N}$$

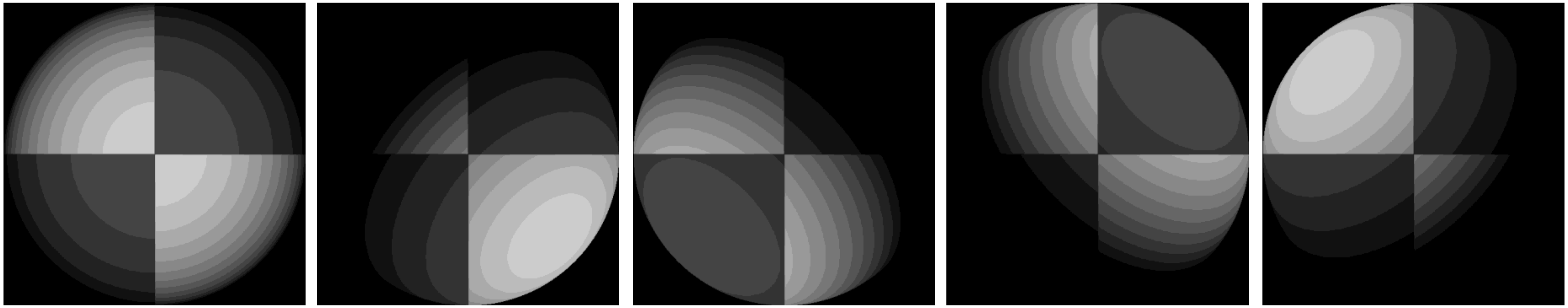
Least squares solution:

$$\begin{aligned} \mathbf{I} &= \mathbf{LG} \\ \mathbf{L^T I} &= \mathbf{L^T L G} \\ \mathbf{G} &= (\mathbf{L^T L})^{-1} (\mathbf{L^T I}) \end{aligned}$$

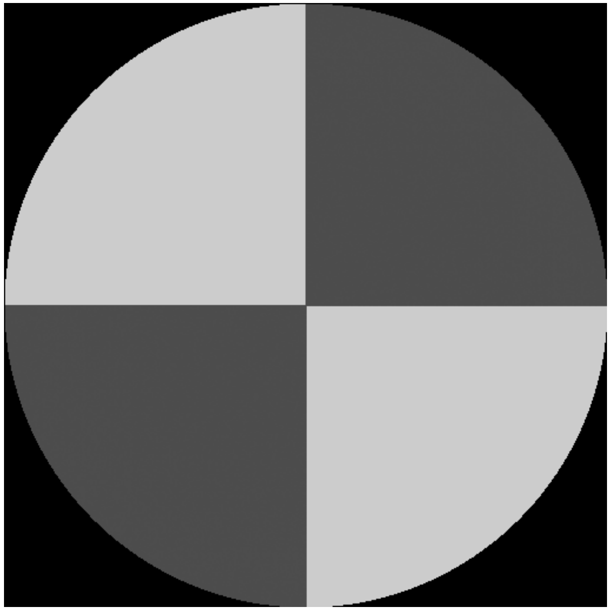
Solve for N, k_d as before

What's the size of $\mathbf{L^T L}$?

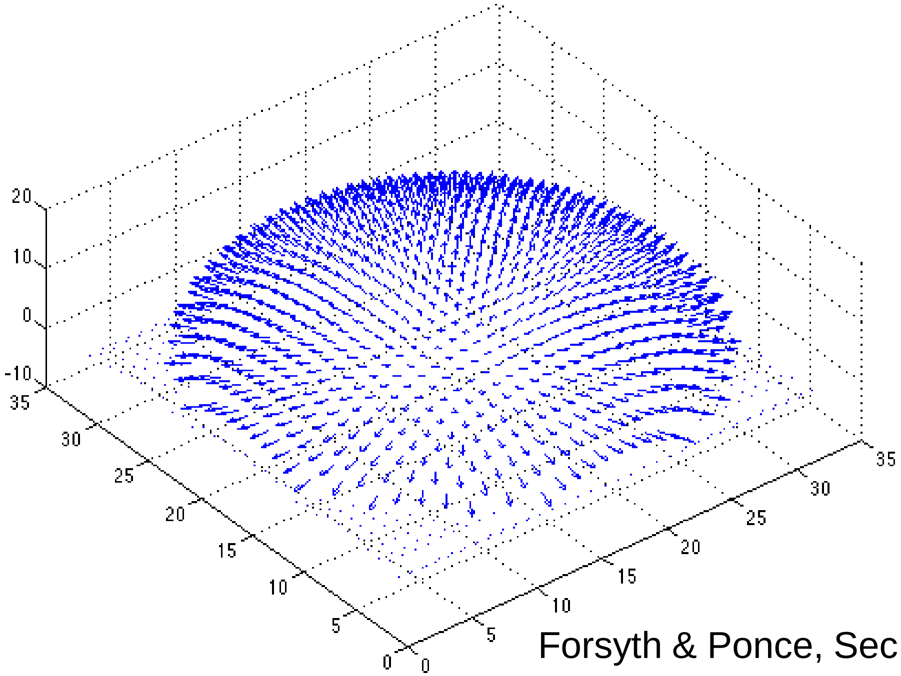
Example



Recovered albedo



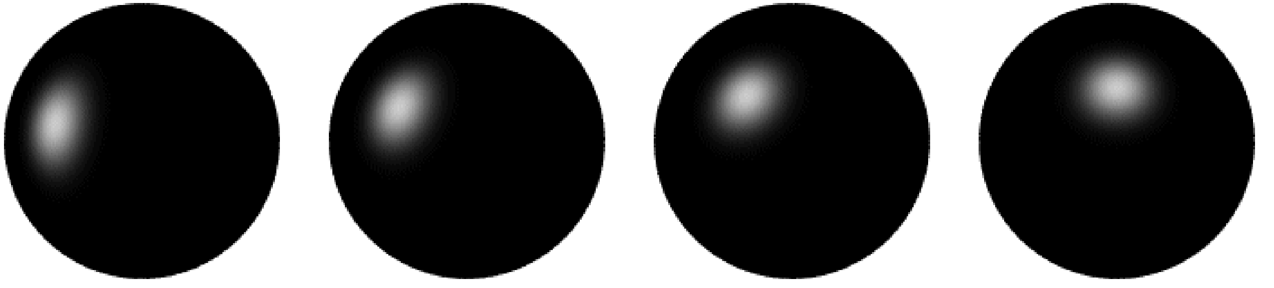
Recovered normal field



Forsyth & Ponce, Sec. 5.4

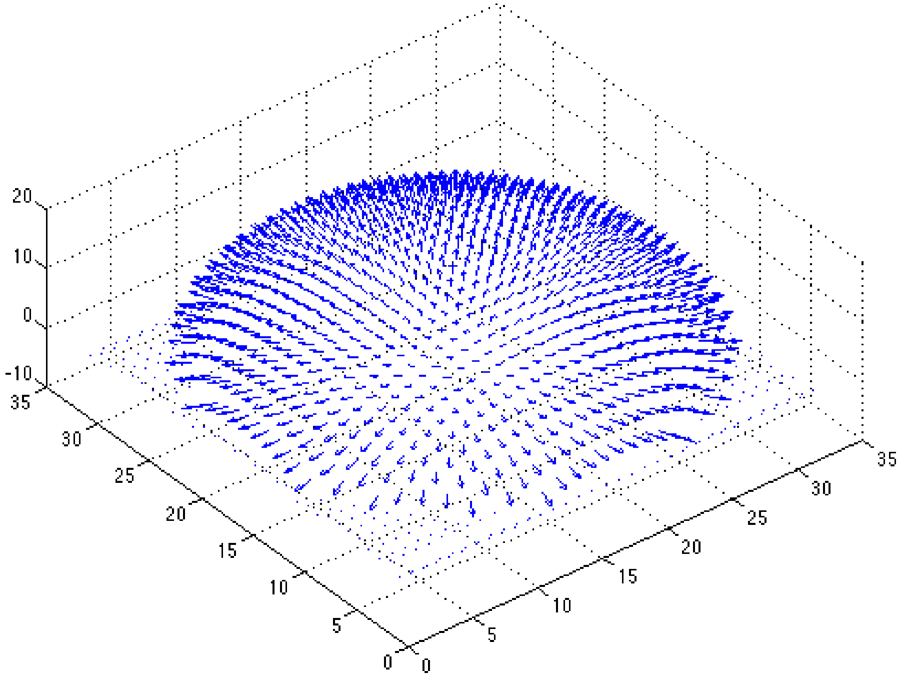
Computing light source directions

Trick: place a chrome sphere in the scene

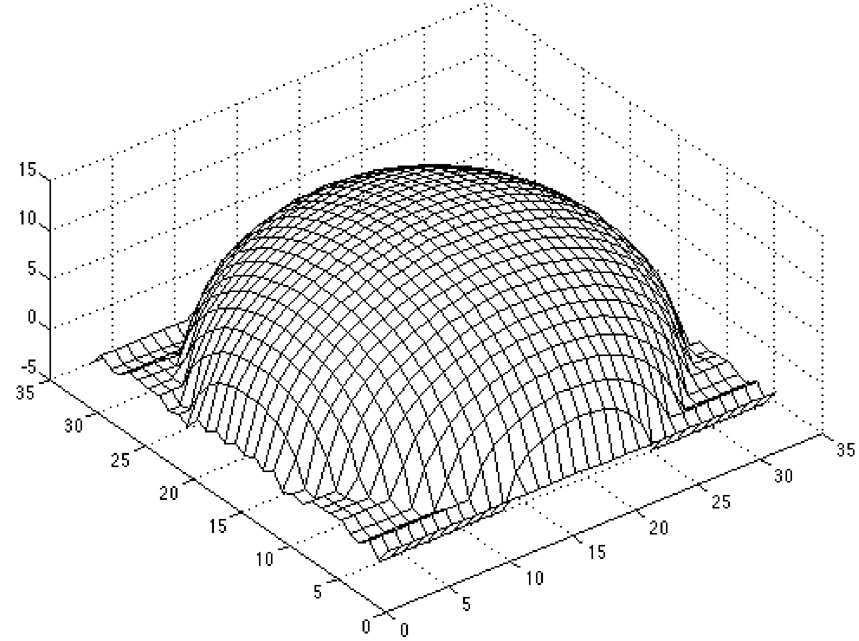


- the location of the highlight tells you where the light source is

Depth from normals



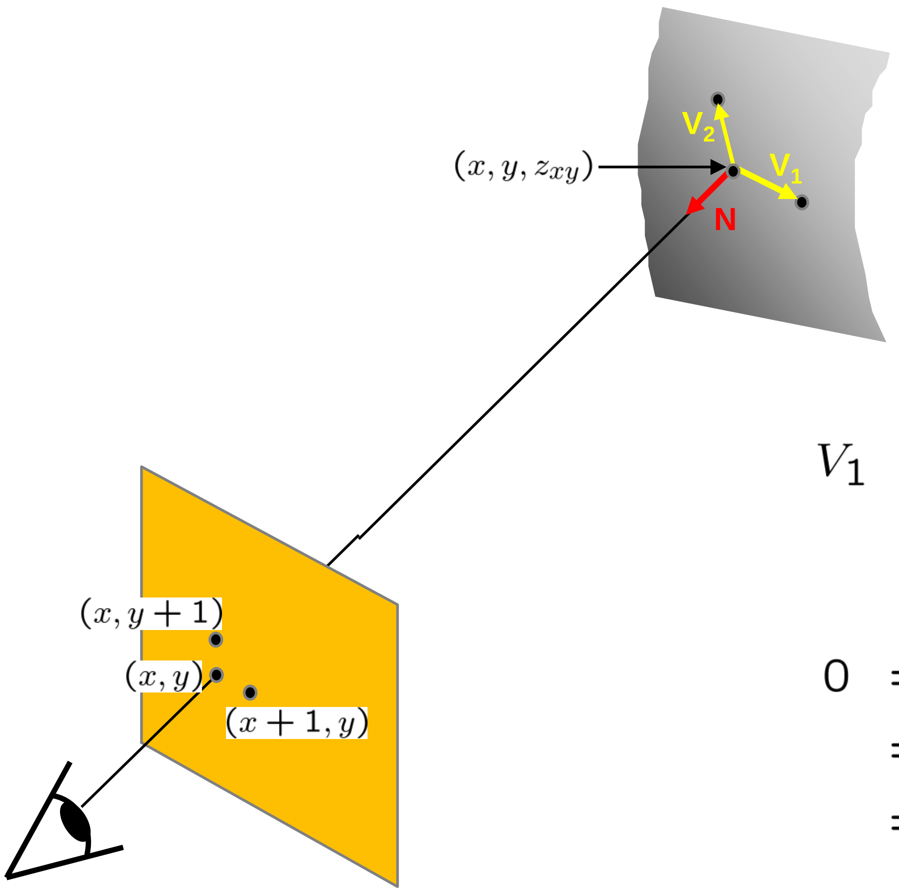
What we have



What we want

Forsyth & Ponce, Sec. 5.4

Depth from normals



$$\begin{aligned} V_1 &= (x+1, y, z_{x+1,y}) - (x, y, z_{xy}) \\ &= (1, 0, z_{x+1,y} - z_{xy}) \end{aligned}$$

$$\begin{aligned} 0 &= N \cdot V_1 \\ &= (n_x, n_y, n_z) \cdot (1, 0, z_{x+1,y} - z_{xy}) \\ &= n_x + n_z(z_{x+1,y} - z_{xy}) \end{aligned}$$

Get a similar equation for $\mathbf{V_2}$

- Each normal gives us two linear constraints on z
- compute z values by solving a matrix equation

Example

Limitations

Big problems

- doesn't work for shiny things, semi-translucent things
- shadows, inter-reflections

Smaller problems

- camera and lights have to be distant
- calibration requirements
  - measure light source directions, intensities
  - camera response function

Newer work addresses some of these issues

Some pointers for further reading:

- Zickler, Belhumeur, and Kriegman, *"Helmholtz Stereopsis: Exploiting Reciprocity for Surface Reconstruction."* IJCV, Vol. 49 No. 2/3, pp 215-227.
- Hertzmann & Seitz, "*Example-Based Photometric Stereo: Shape Reconstruction with General, Varying BRDFs*." IEEE Trans. PAMI 2005

Finding the direction of the light source

P. Nillius and J.-O. Eklundh, "Automatic estimation of the projected light source direction," CVPR 2001

Application: Detecting composite photos

Real photo?

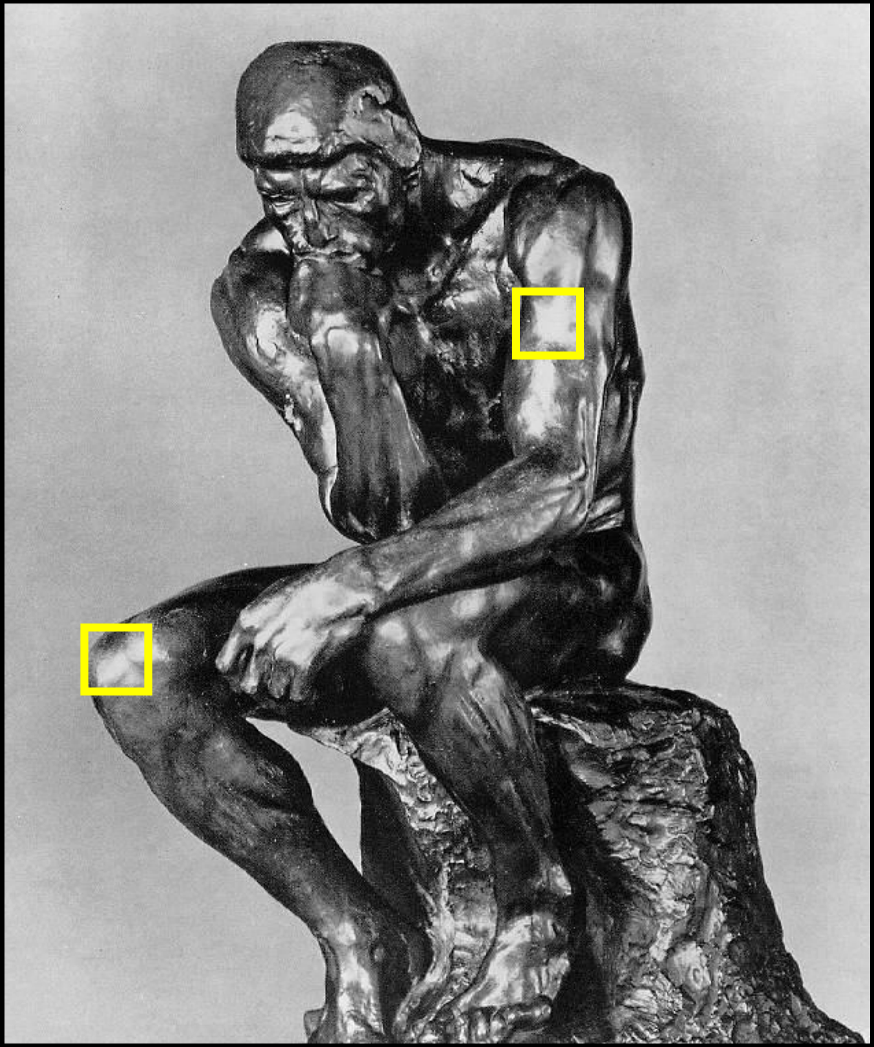
Fake photo

Example-based Photometric Stereo

Aaron Hertzmann
University of Toronto

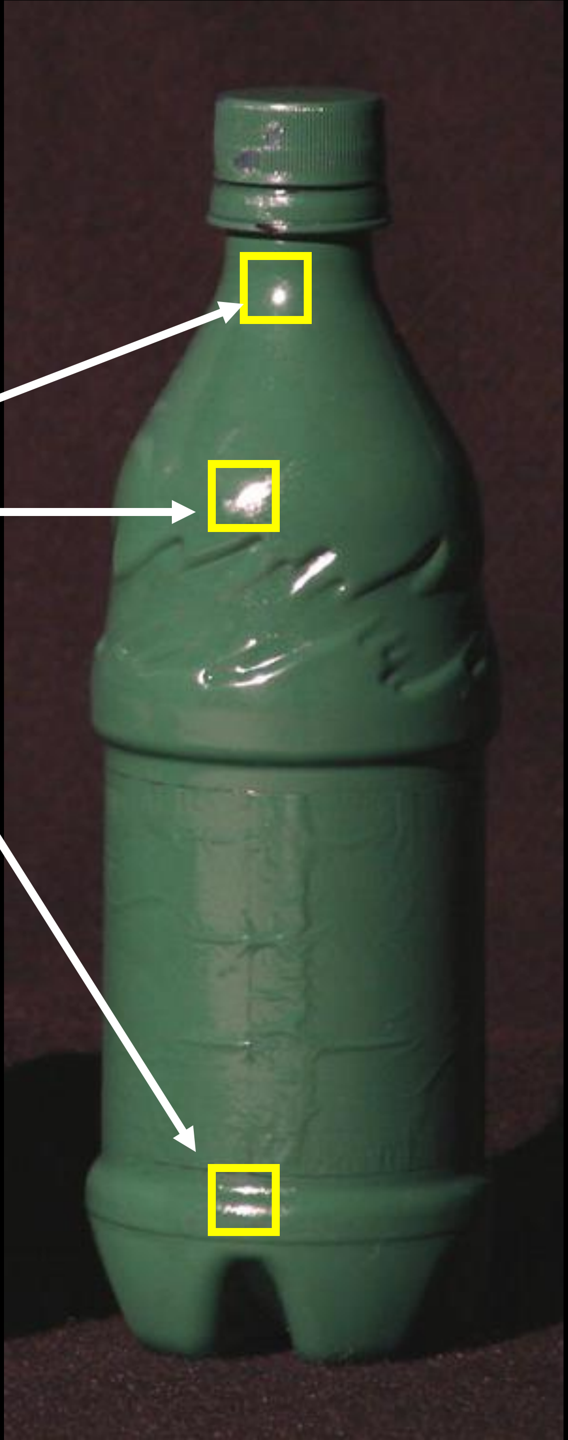
Steven M. Seitz
University of Washington

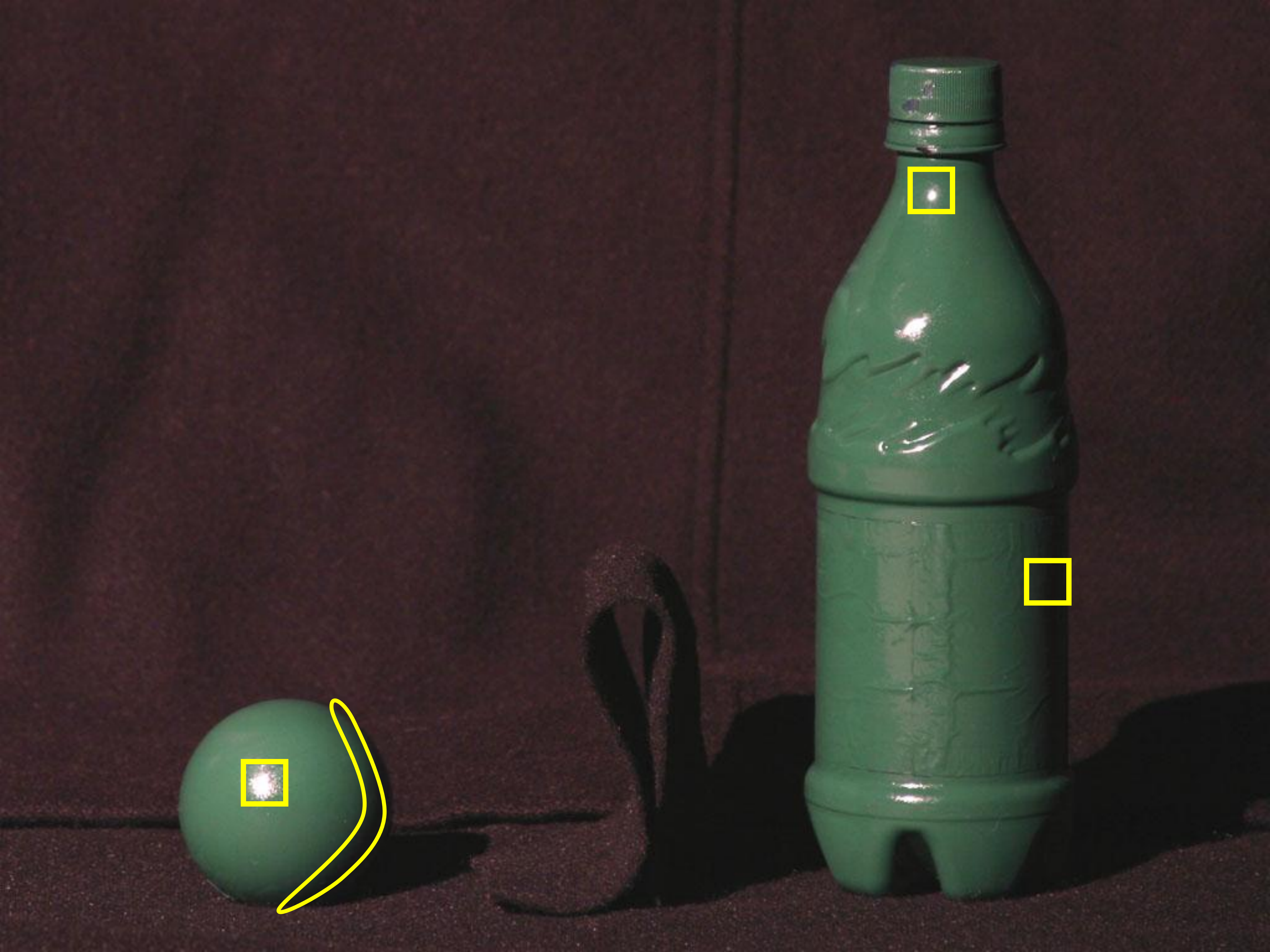
Shiny things

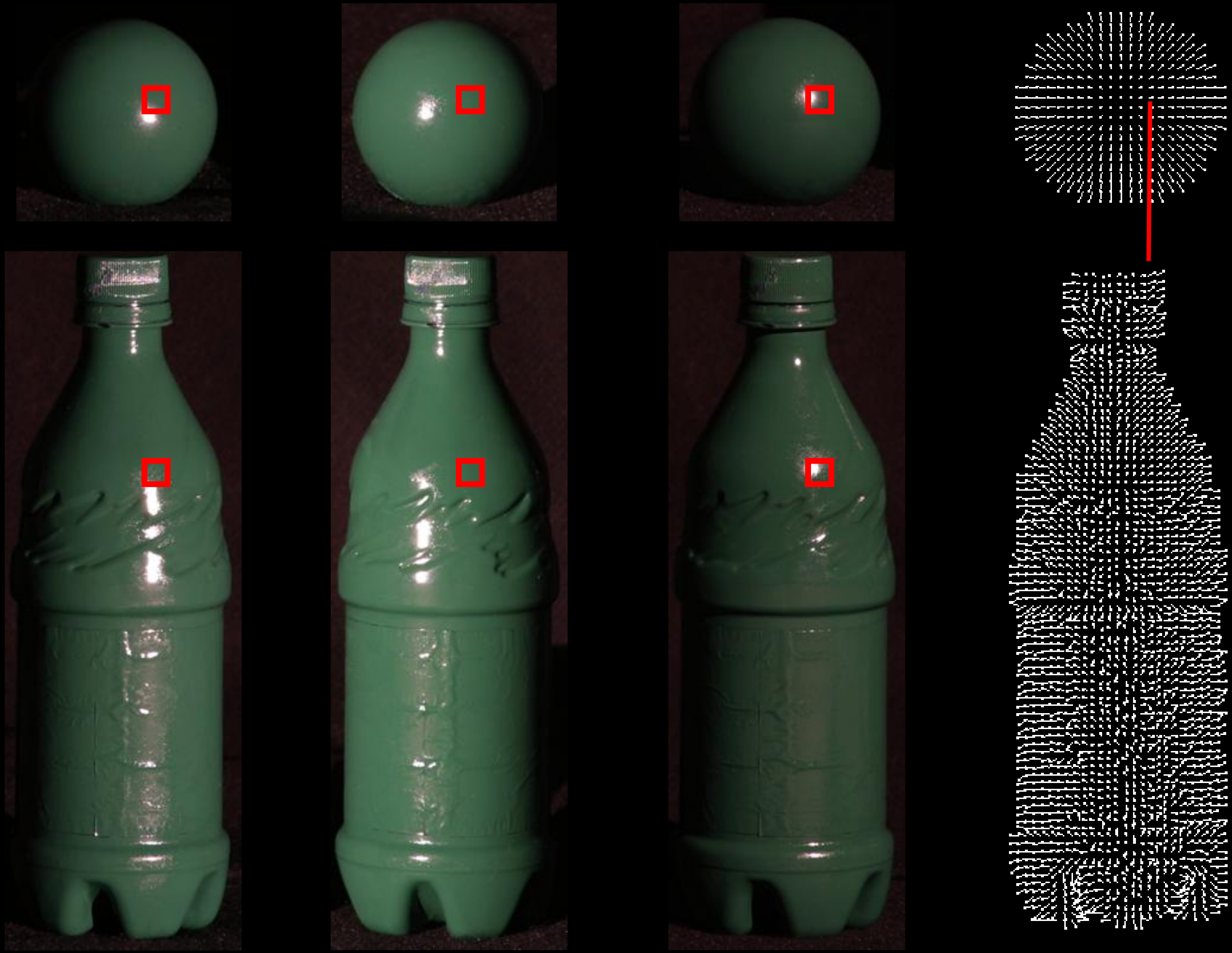


"Orientation consistency"

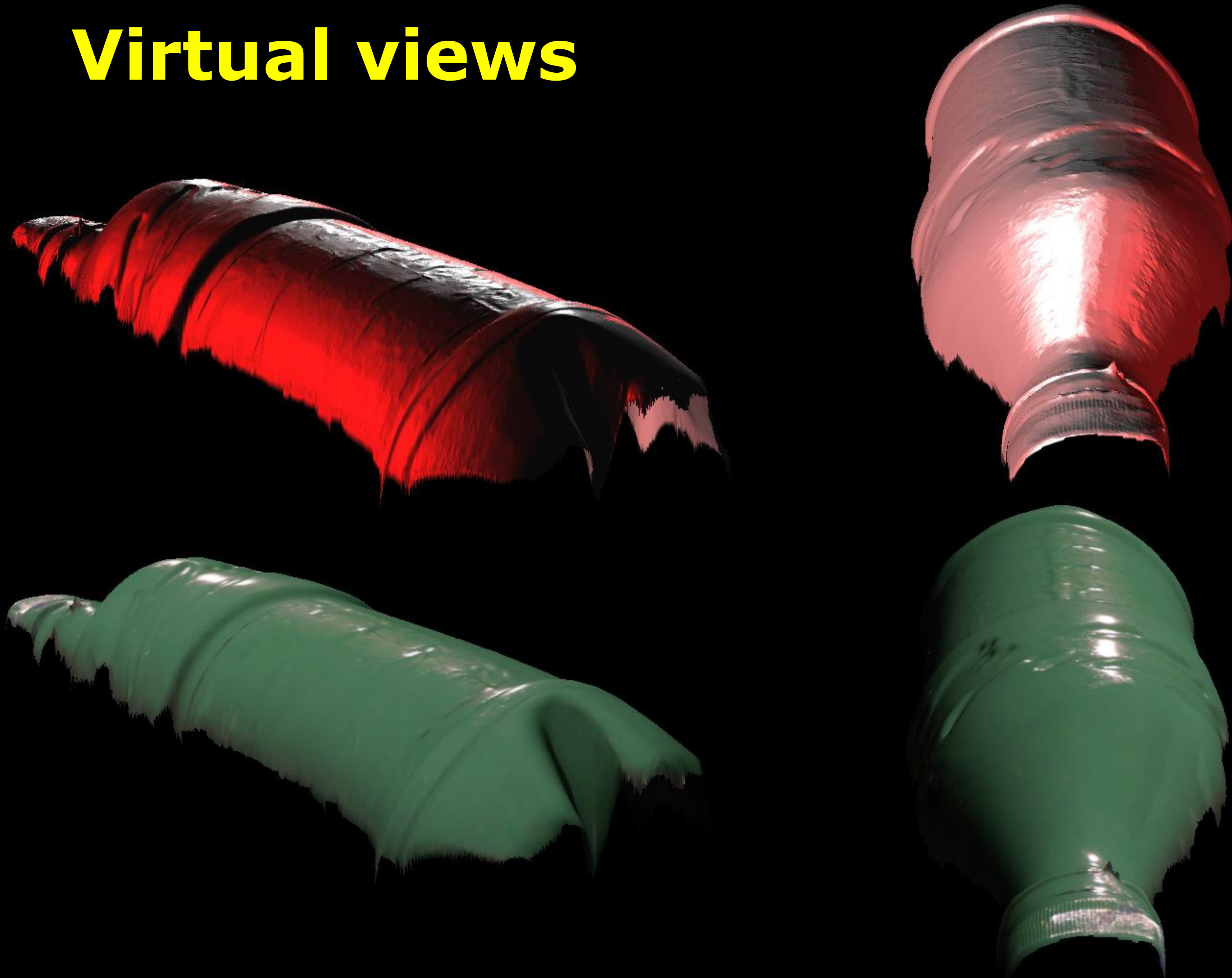
same surface normal







Virtual views

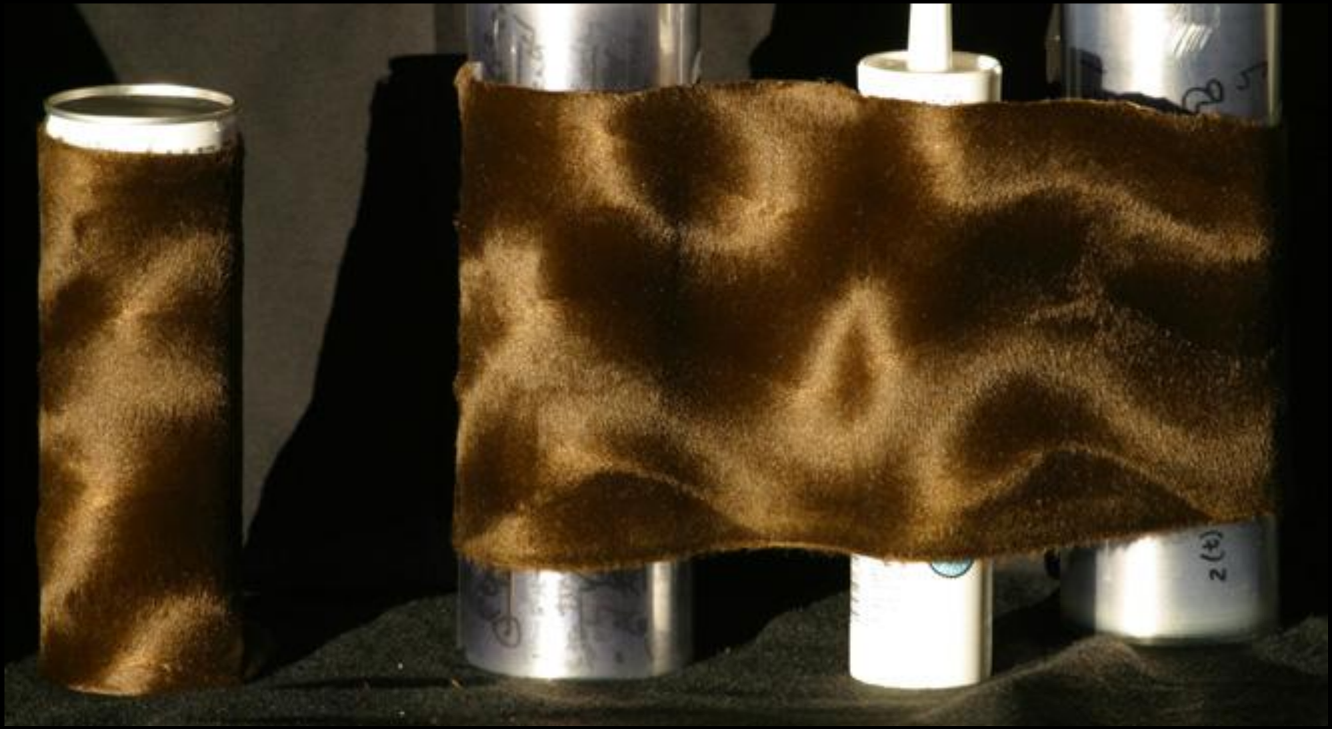


Velvet

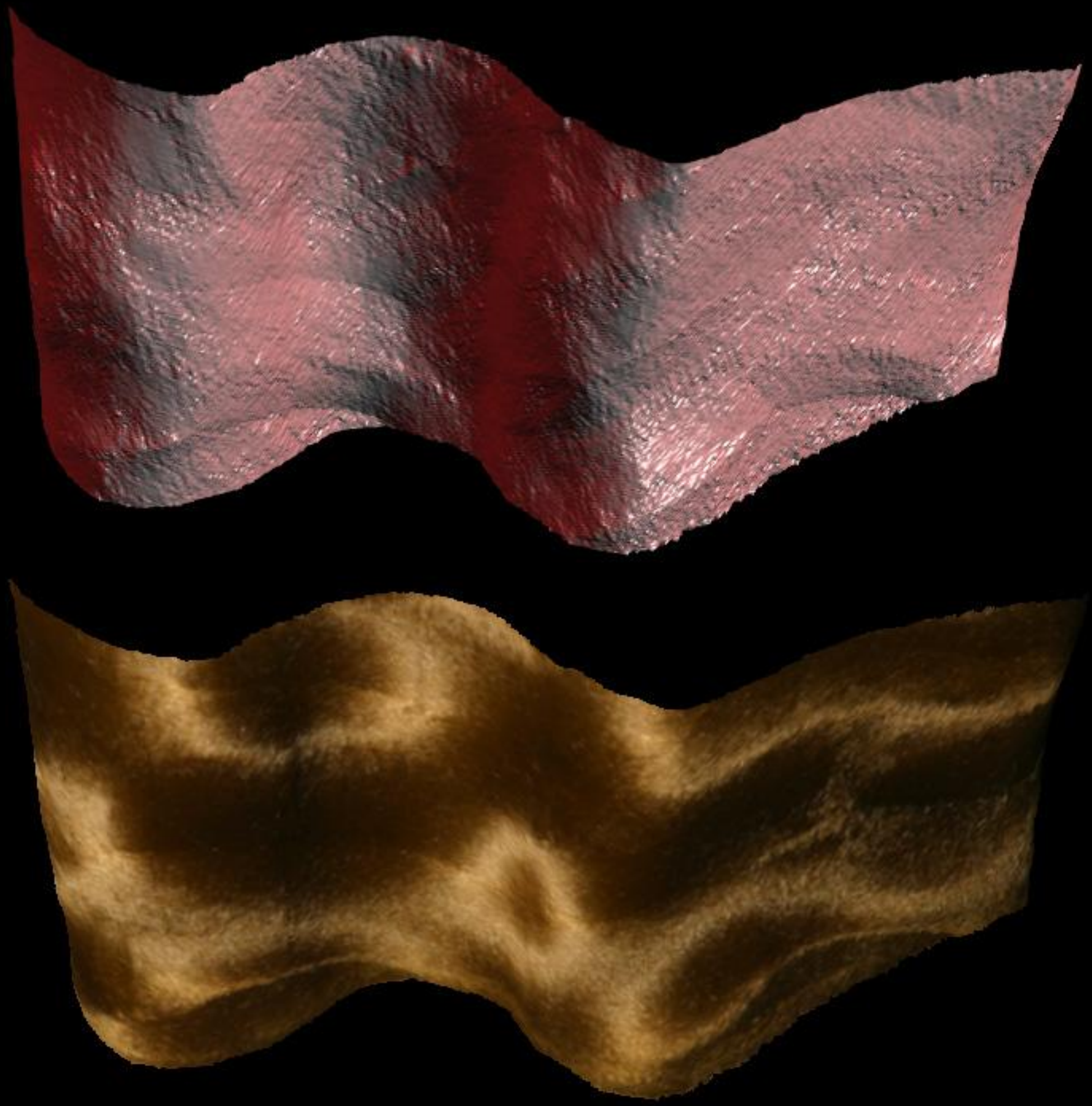
Virtual Views

Brushed Fur

Brushed Fur



Virtual Views



Salem Specialty Ball Company

Salem Specialty Ball supplies industrial grade balls that are used in bearings, pumps, valves and other commercial applications. We can supply balls in just about any size that is machineable. We have produced precision balls from .002" all the way up to 12.0" and beyond. We can also produce these balls in any material. Almost without exception, if the material exists, we can make it into a ball. Not only do we specialize in hard to find materials, we also carry standard materials such as chrome steel and the stainless steels. We stock an extensive inventory of ready to ship balls. Most orders are shipped the same day. And if is isn't in stock, we can make it for you in matter of days. In addition, you will find that our prices are very competitive.

Located in the beautiful northwest corner of Connecticut, Canton has been our company's home for the last three years and we have been in complete operation for over ten years. Proud of our reputation, Salem Specialty Ball Company has over fifty years of combined experience allowing us to provide top-notch quality technical support and expert engineering consultation

Questions?

3-minute break

Computational Photography

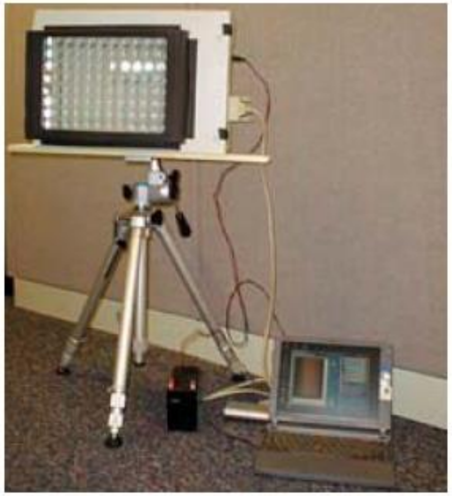


Image from Durand & Freeman's MIT Course on Computational Photography

Today's reading

- Szeliski Chapter 9

The ultimate camera

What does it do?

The ultimate camera

Infinite resolution

Infinite zoom control

Desired object(s) are in focus

No noise

No motion blur

Infinite dynamic range (can see dark and bright things)

...

Creating the ultimate camera

The “analog” camera has changed very little in >100 yrs

- we’re unlikely to get there following this path

More promising is to combine “analog” optics with computational techniques

- “Computational cameras” or “Computational photography”

This lecture will survey techniques for producing higher quality images by combining optics and computation

Common themes:

- take multiple photos
- modify the camera

Noise reduction

Take several images and average them

Why does this work?

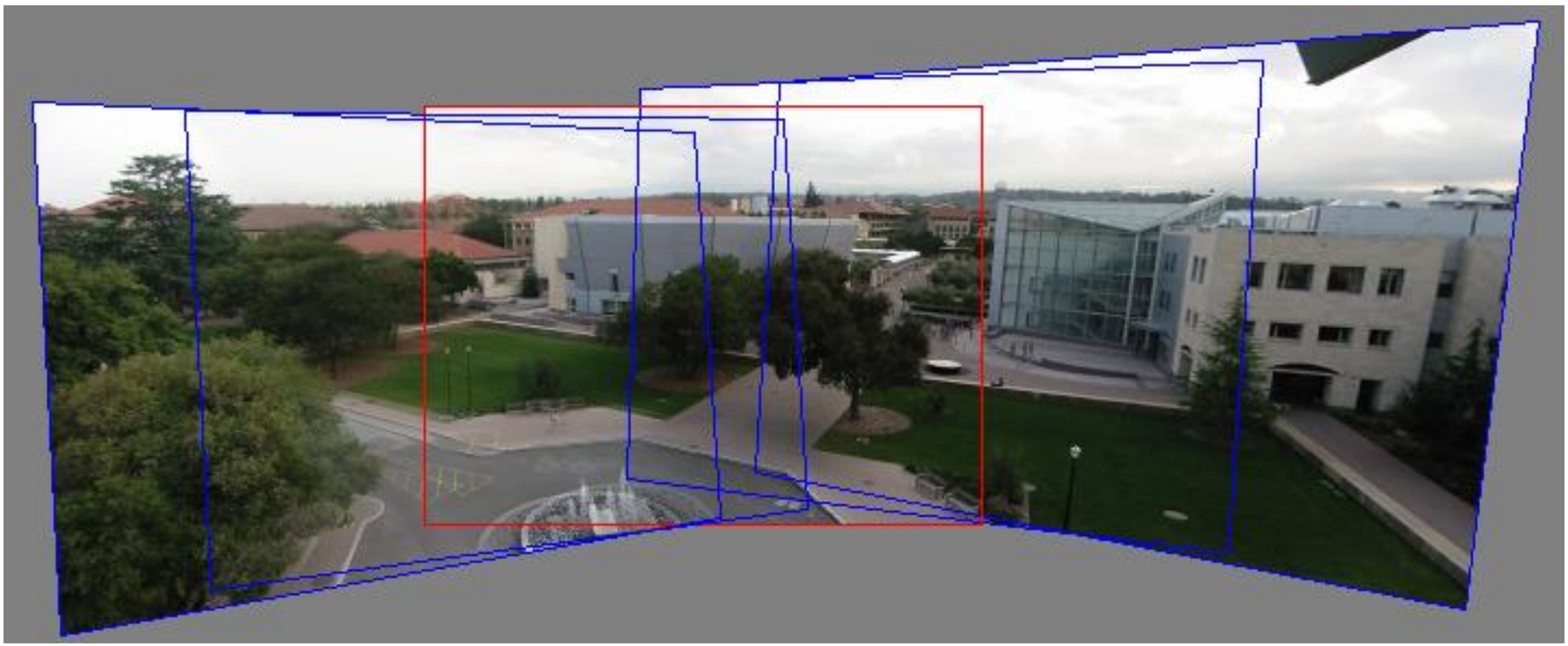
Basic statistics:

- variance of the mean decreases with n:

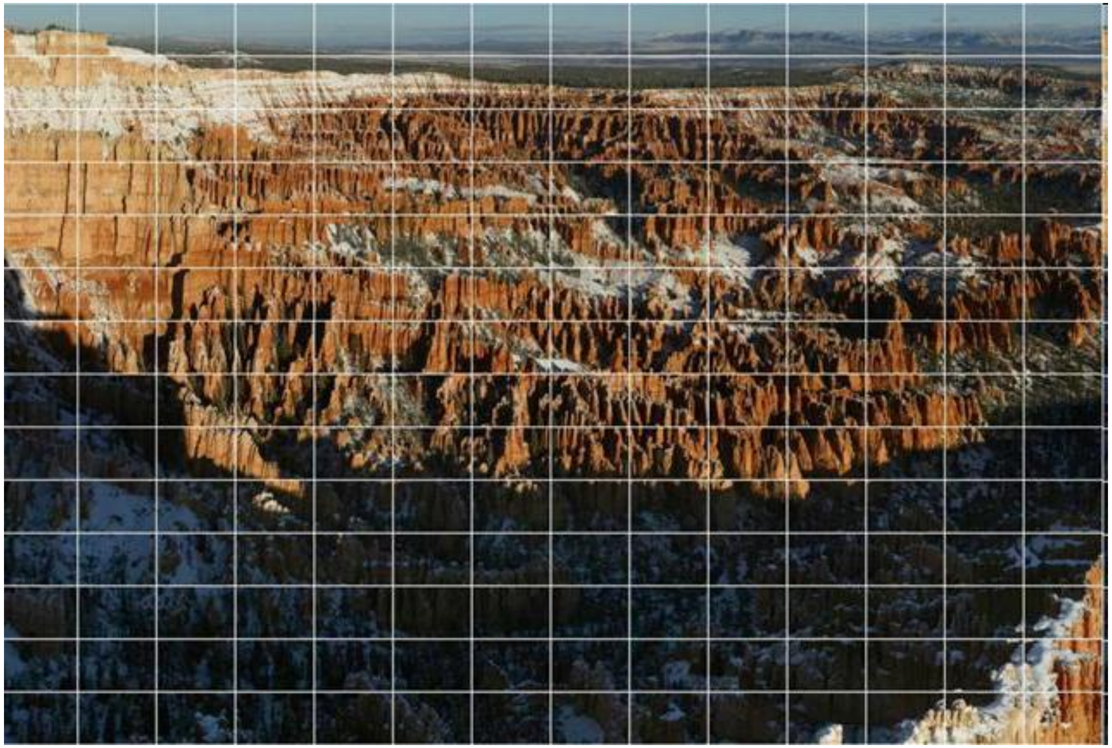
$$Var(\overline{X}) = \frac{\sigma^2}{n}$$

Field of view

We can artificially increase the field of view by compositing several photos together (project 2).



Improving resolution: Gigapixel images



Max Lyons, 2003
fused 196 telephoto shots

A few other notable examples:

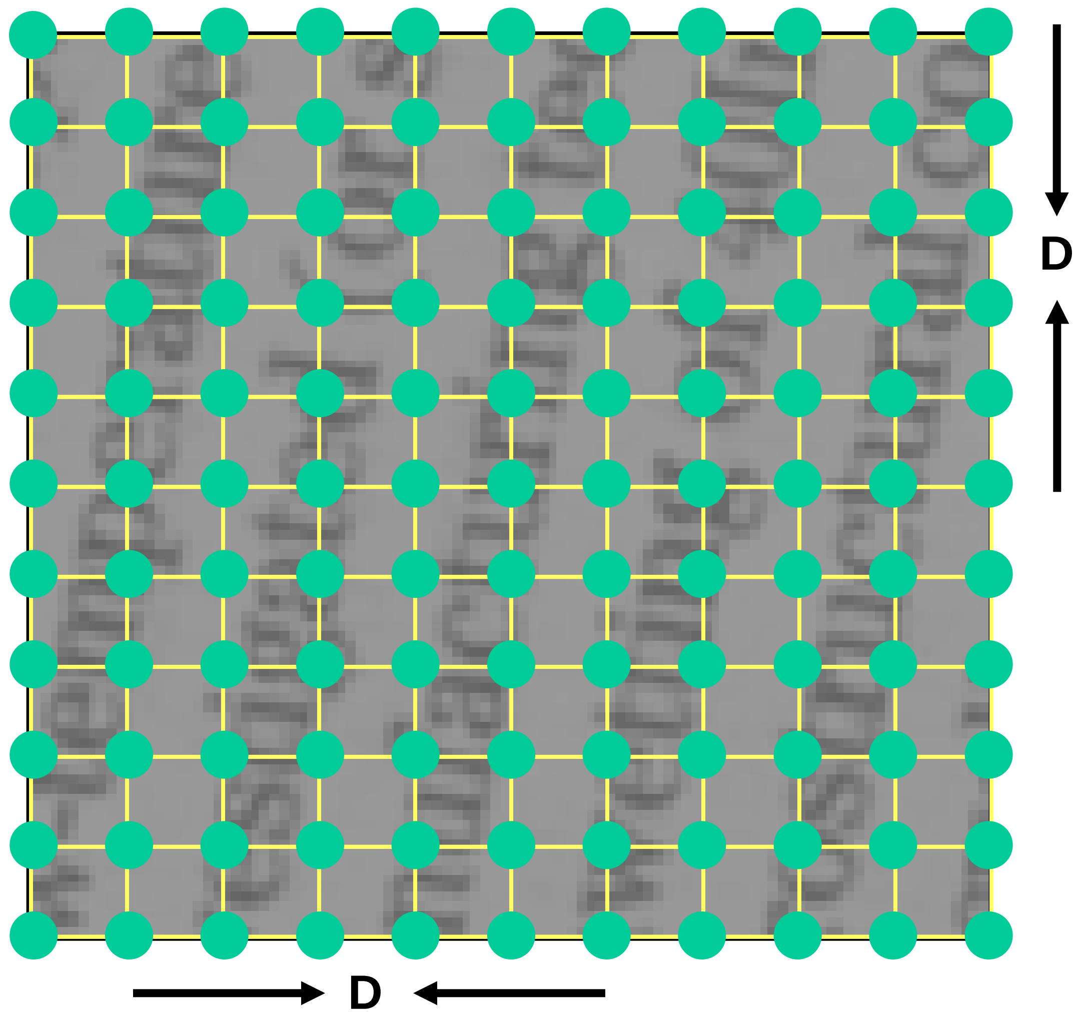
- Obama inauguration (gigapan.org)
- HDView (Microsoft Research)

Improving resolution: super resolution

What if you don’t have a zoom lens?

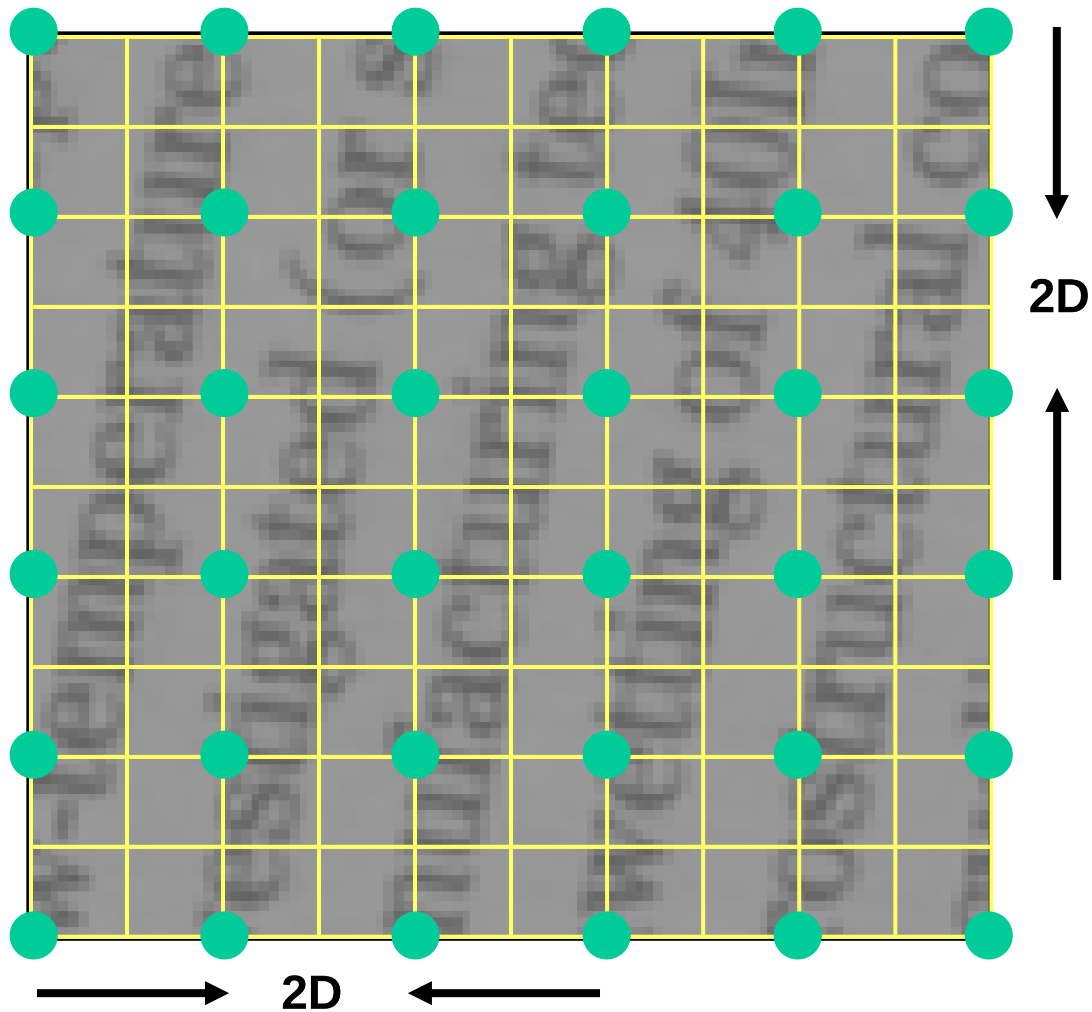
Intuition (slides from Yossi Rubner & Miki Elad)

For a given band-limited image, the Nyquist sampling theorem states that if a uniform sampling is fine enough ($\geq$**D**), perfect reconstruction is possible.



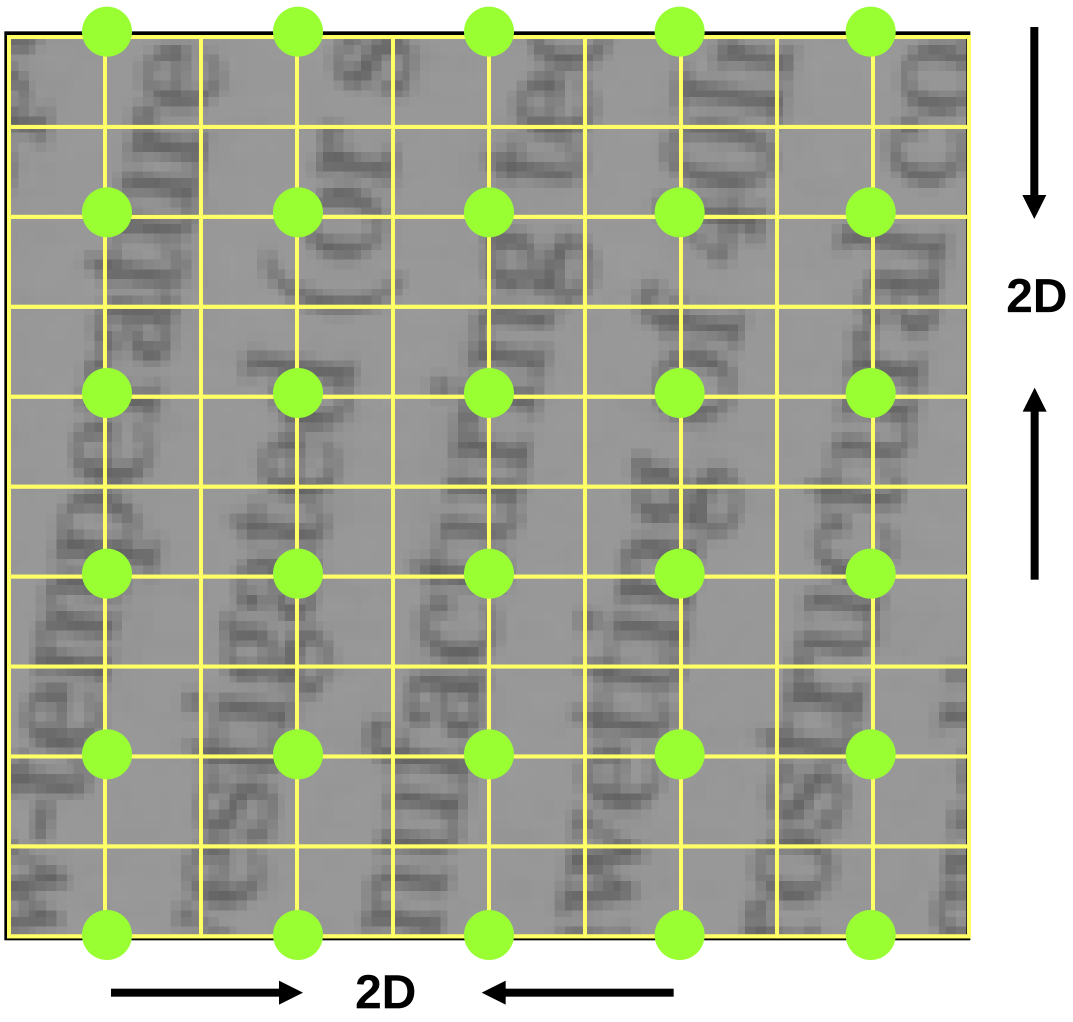
Intuition (slides from Yossi Rubner & Miki Elad)

Due to our limited camera resolution, we sample using an insufficient 2D grid



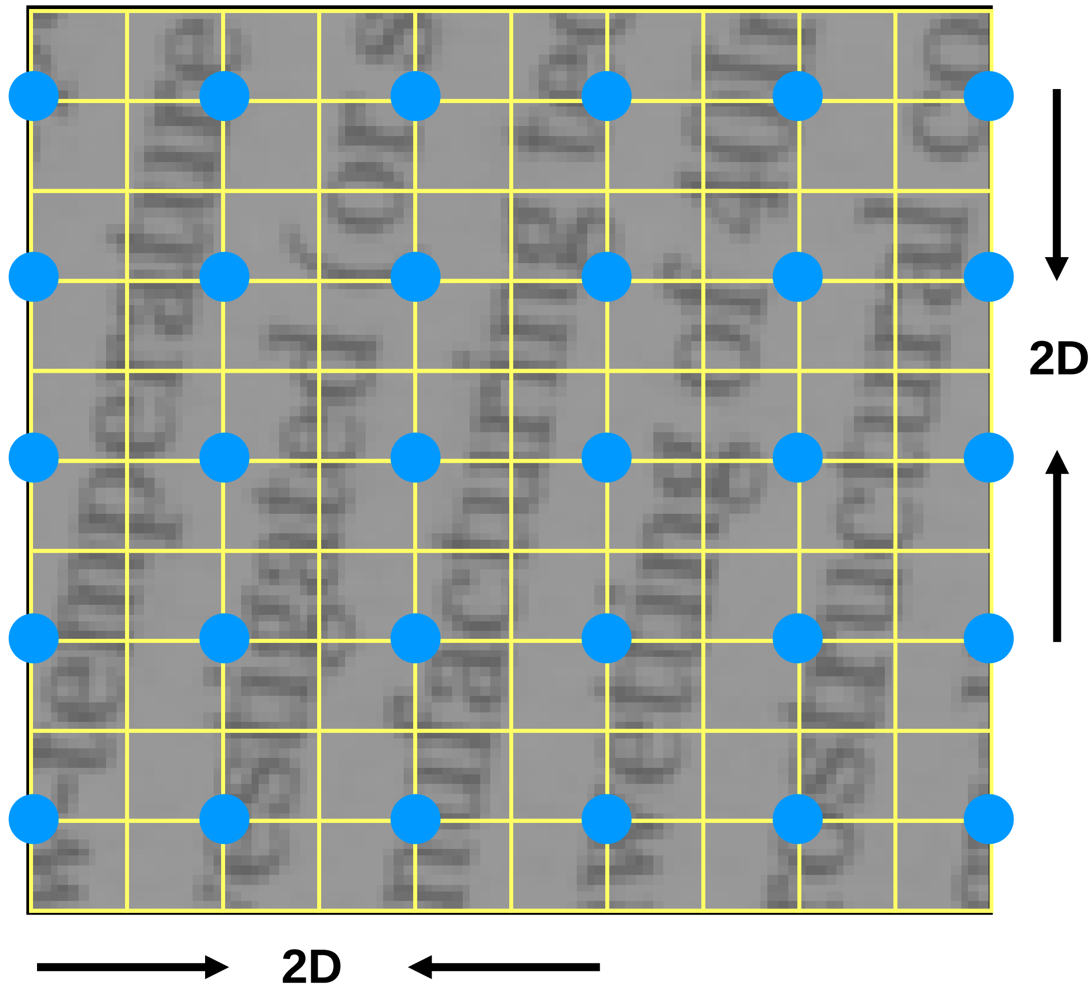
Intuition (slides from Yossi Rubner & Miki Elad)

However, if we take a second picture, shifting the camera 'slightly to the right' we obtain:



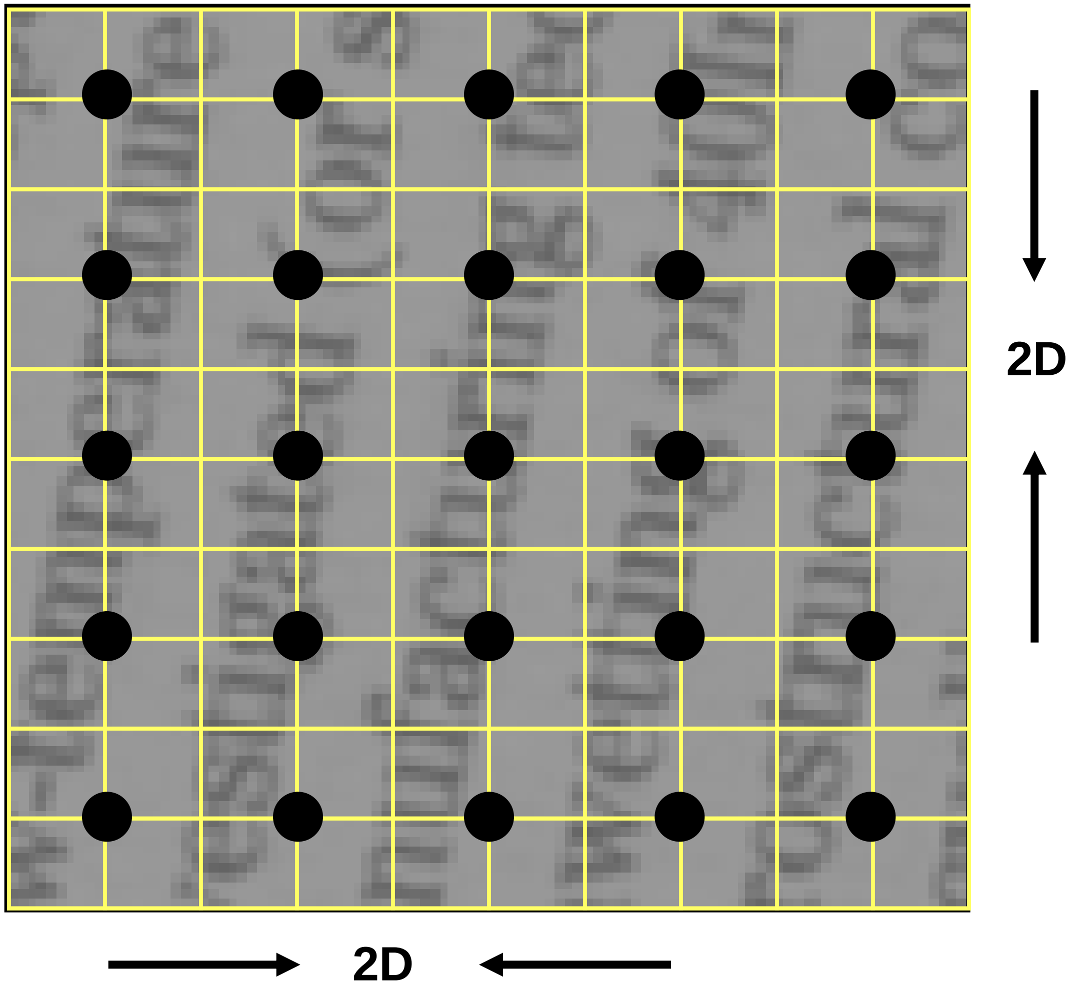
Intuition (slides from Yossi Rubner & Miki Elad)

Similarly, by shifting down we get a third image:



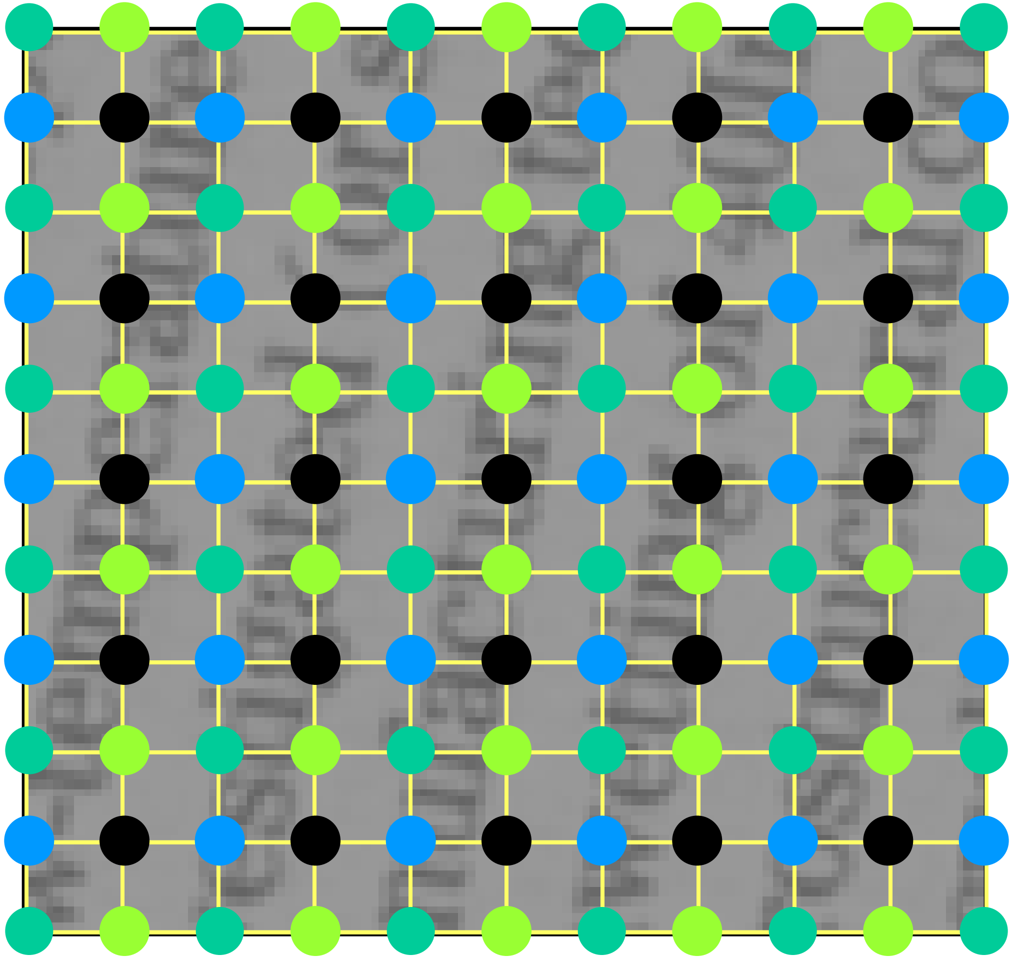
Intuition (slides from Yossi Rubner & Miki Elad)

And finally, by shifting down and to the right we get the fourth image:

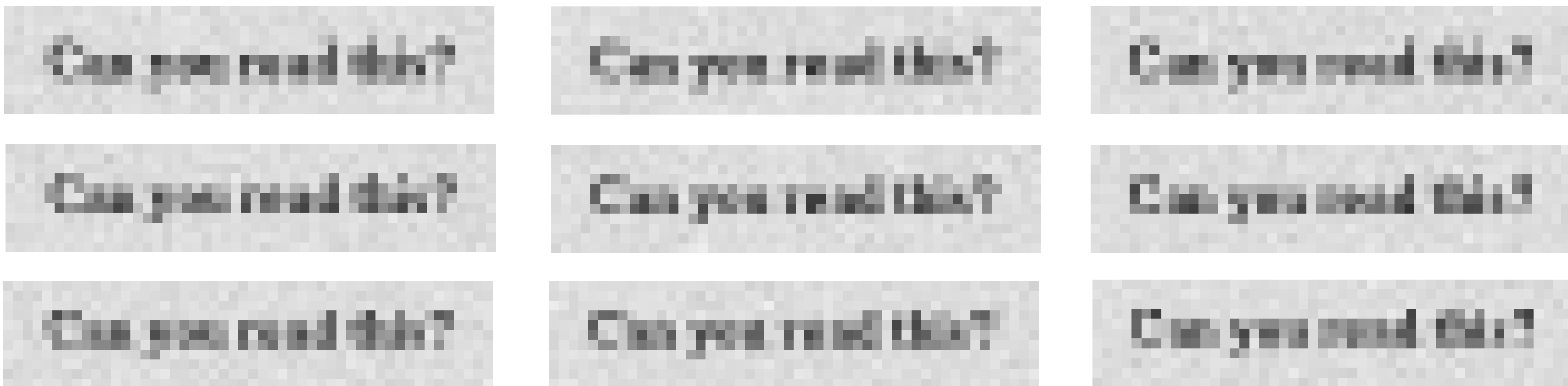


Intuition

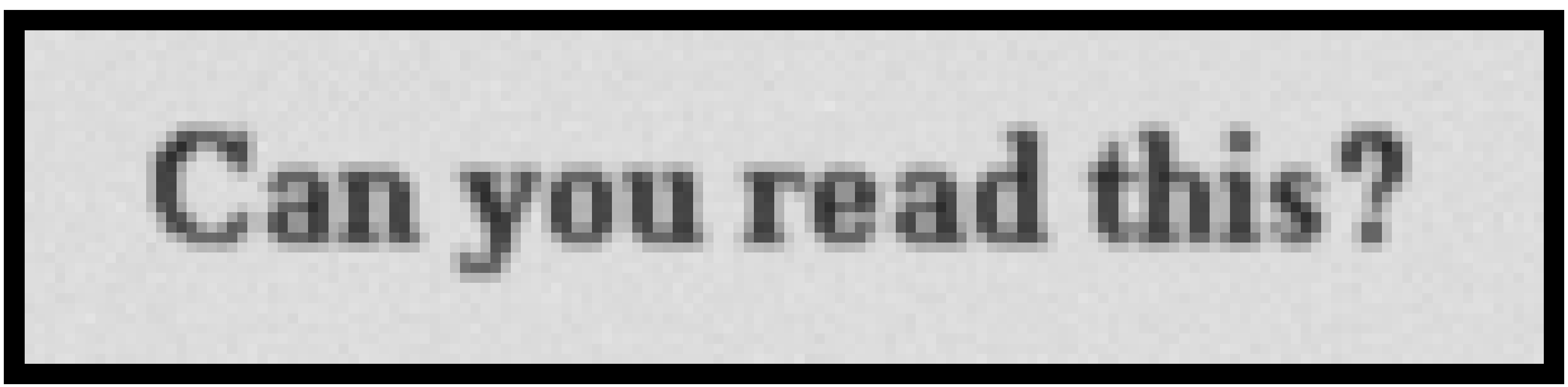
By combining all four images the desired resolution is obtained, and thus perfect reconstruction is guaranteed.



Example

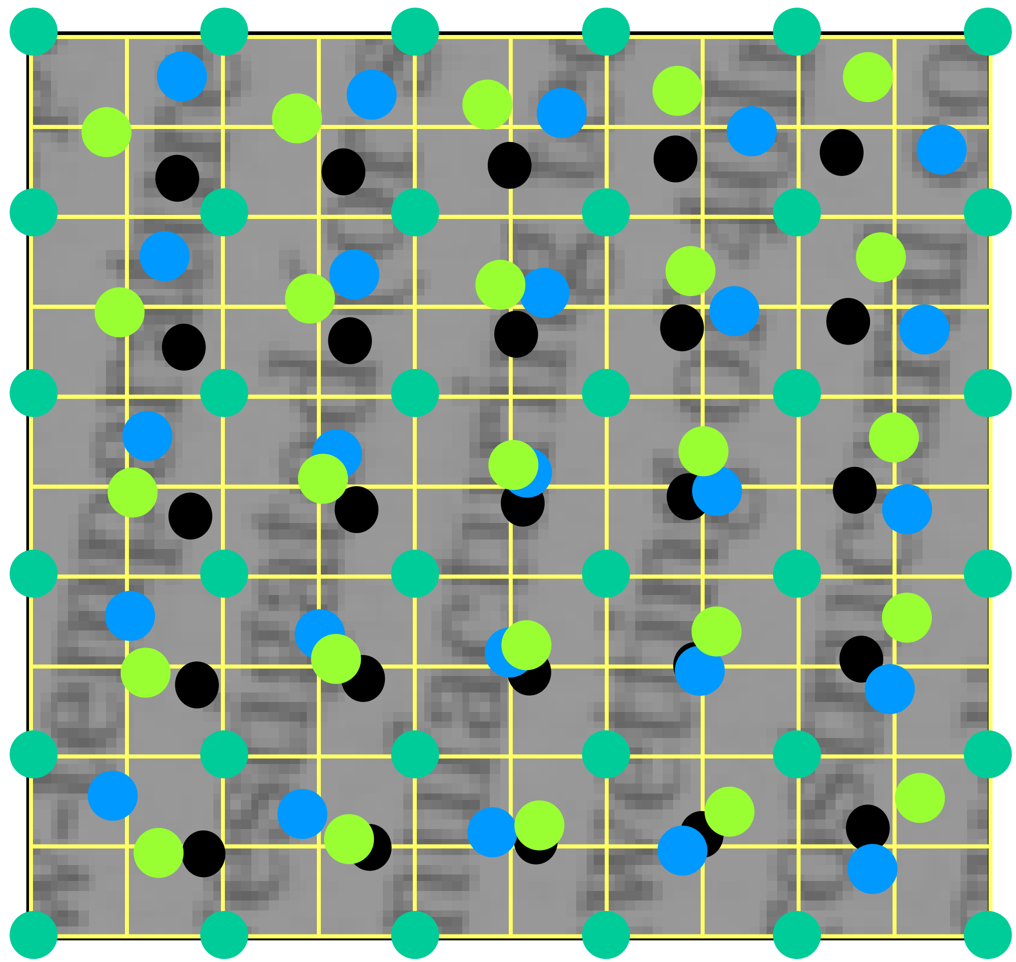


3:1 scale-up in each axis using 9 images, with pure global translation between them



Handling more general 2D motions

What if the camera displacement is Arbitrary ?
What if the camera rotates? Gets closer to the object (zoom)?



Super-resolution

Basic idea:

- define a destination (dst) image of desired resolution
- assume mapping from dst to each input image is known
  - usually a combination of a 2D motion/warp and an average (point-spread function)
  - can be expressed as a set of linear constraints
  - sometimes the mapping is solved for as well
- add some form of regularization (e.g., “smoothness assumption”)
  - can also be expressed using linear constraints
  - but L1, other nonlinear methods work better

How does this work? [Baker & Kanade, 2002]

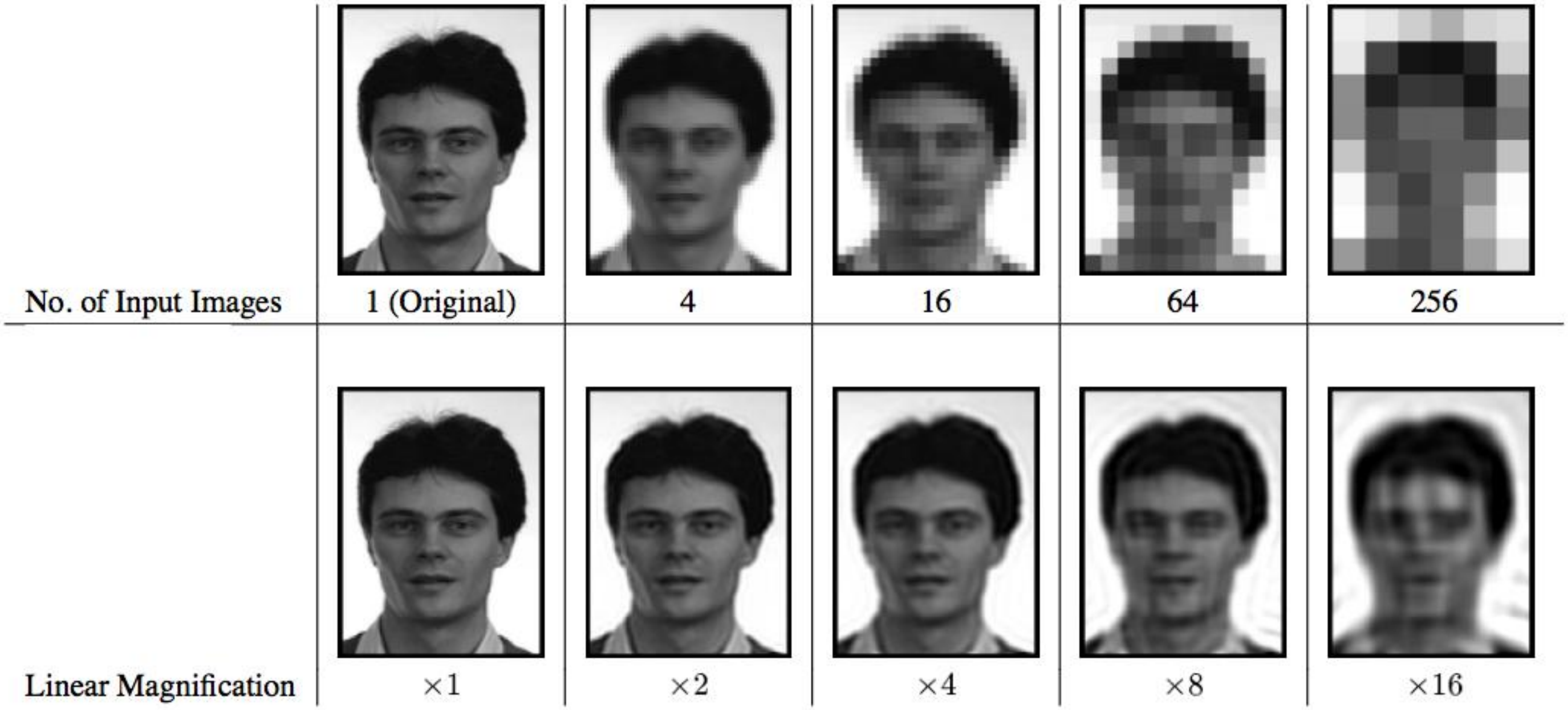


Figure 1: Results of the reconstruction-based super-resolution algorithm [9] for increasing magnification factors. The original high-resolution image (top left) is translated multiple times by random sub-pixel amounts, blurred with a Gaussian, and then down-sampled. Comparing the images in the right-most column, we see that the reconstruction algorithm does quite well given the very low resolution of the input. The degradation in performance as the magnification increases from left to right is very dramatic, however.

Limits of super-resolution [Baker & Kanade, 2002]

Performance degrades significantly beyond 4x or so

Doesn’t matter how many new images you add

- space of possible (ambiguous) solutions explodes quickly

Major cause

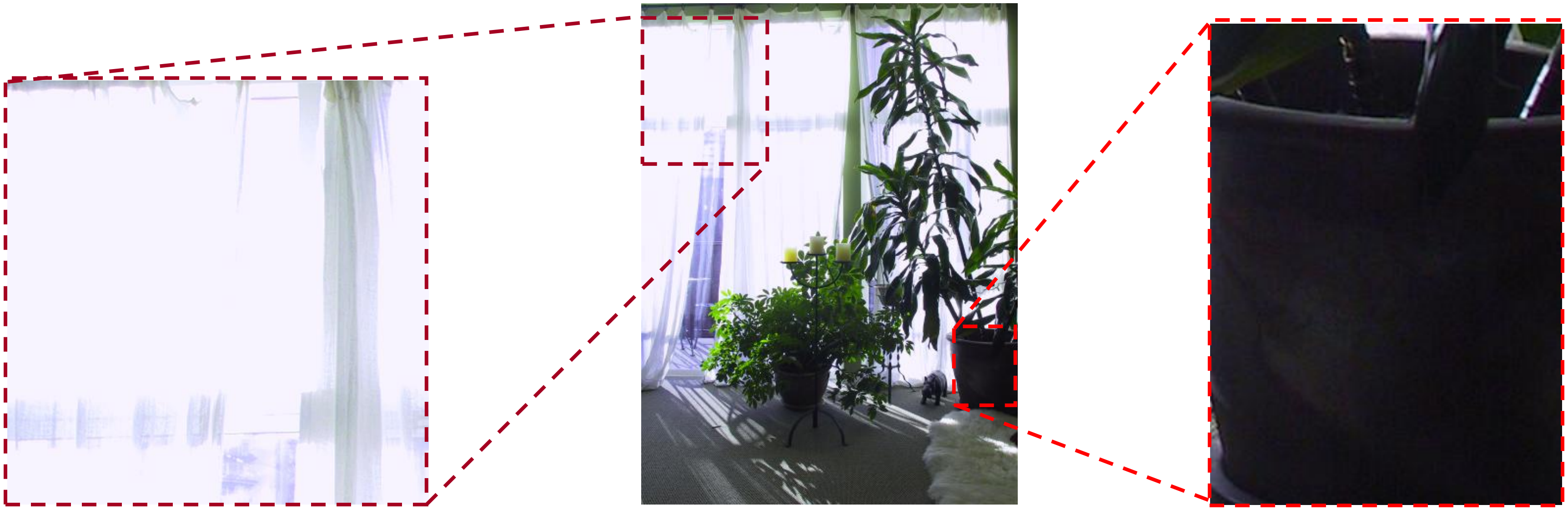
- quantizing pixels to 8-bit gray values

Possible solutions:

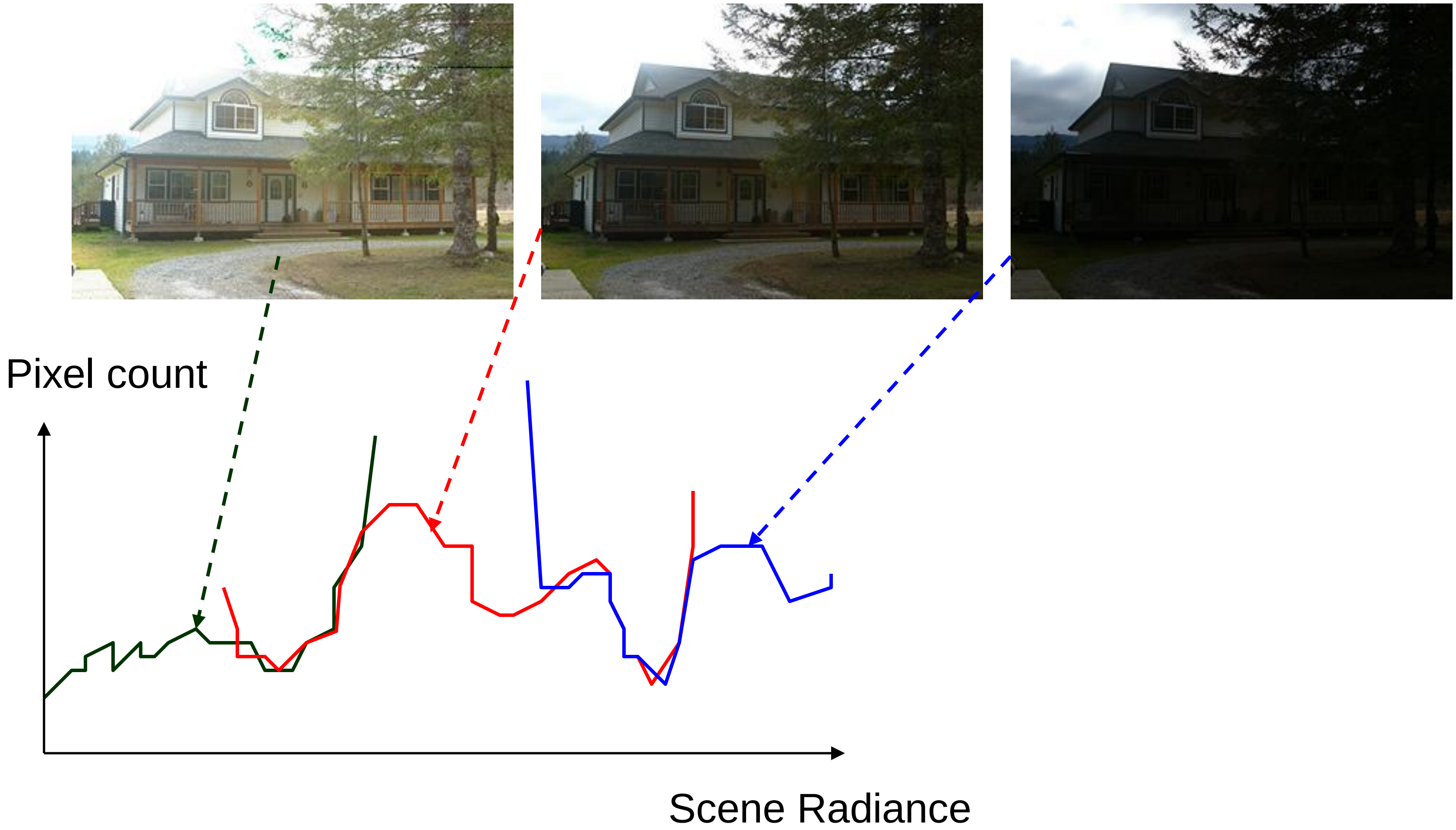
- nonlinear techniques (e.g., L1)
- better priors (e.g., using domain knowledge)
  - Baker & Kanade “Hallucination”, 2002
  - Freeman et al. “Example-based super-resolution”

Dynamic Range

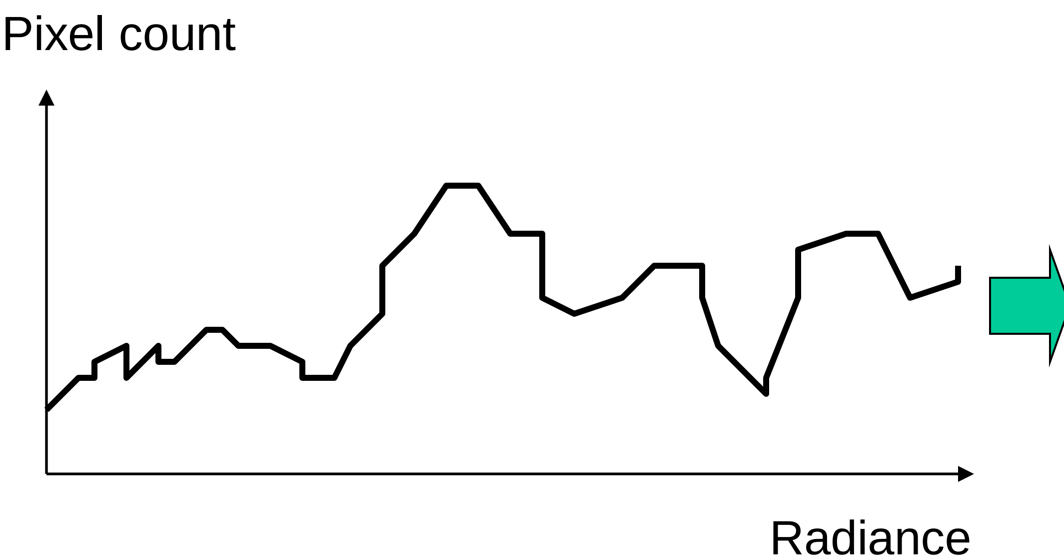
Typical cameras have limited dynamic range



HDR images — merge multiple inputs



HDR images — merged



Camera is not a photometer!

Limited dynamic range

- 8 bits captures only 2 orders of magnitude of light intensity
- We can see ~10 orders of magnitude of light intensity

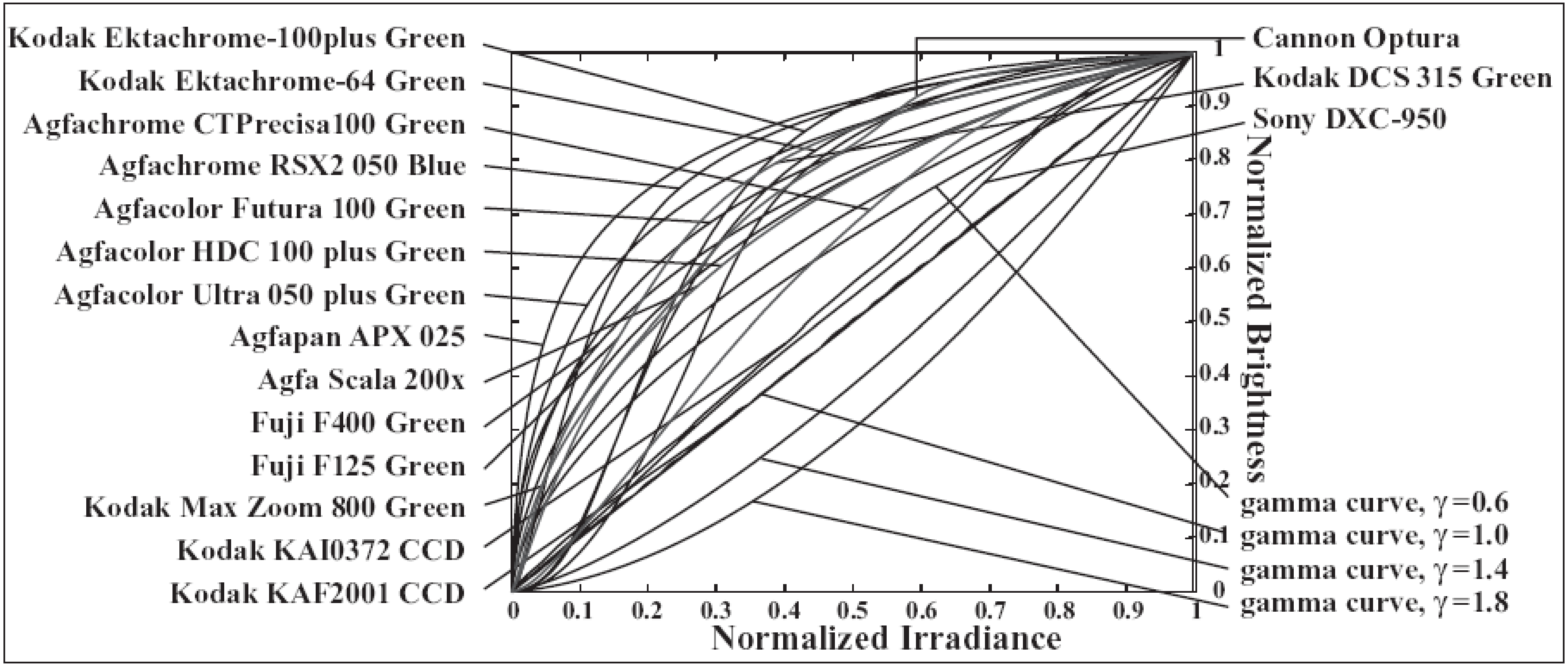
Unknown, nonlinear response

- pixel intensity $\neq$ amount of light (# photons, or "radiance")

Solution:

- Recover response curve from multiple exposures, then reconstruct the ***radiance map***

Camera response function



Capture and composite several photos

Same trick works for

- field of view
- resolution
- signal to noise
- dynamic range
- Focus

But sometimes you can do better by modifying the camera…