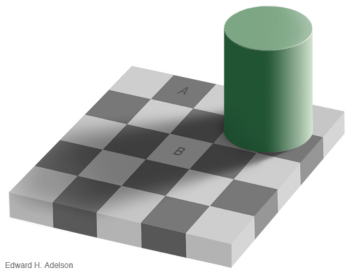
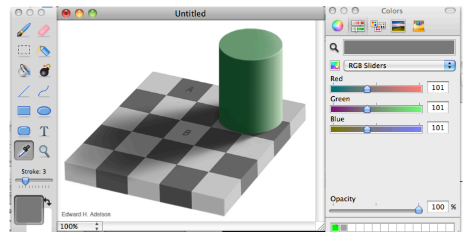
Lighting and Reflectance



The squares marked A and B are the same shade of gray

Lighting



Let's go to paintbrush

Lighting

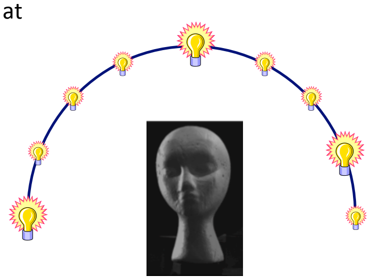
Lighting can have a big effect on how an object looks.

Modeling the effect of lighting can be used for:

Recognition – particularly face recognition
Shape reconstruction
Motion estimation
Re-rendering / Re-lighting
…

Lighting is Complex

- Lighting can come from any direction and at any strength
- Infinite degree of freedom

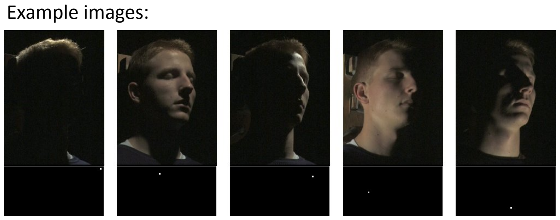


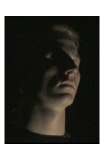
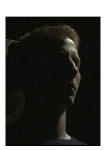
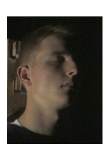
Capture lighting variation

Illuminate subject from many incident directions

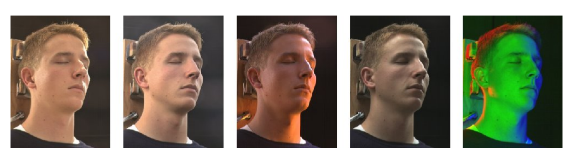
From Ravi Ramamoorthi

Example images:



 = .2  + .3  + ...

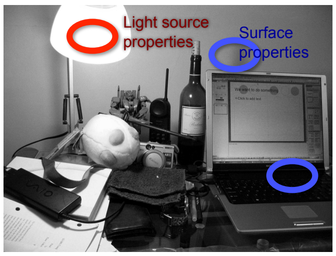
From Ravi Ramamoorthi



From Ravi Ramamoorthi

Image brightness

What determines the brightness of an image pixel?



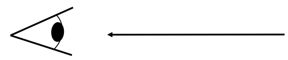
- lighting
- Surface BRDF (local reflectance)
- Shadowing
- Inter-reflections (global reflectance)

What is light?

Electromagnetic radiation (EMR) moving along rays in space

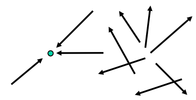
- R(λ) is EMR, measured in units of power (watts)

– λ is wavelength



Light field

- We can describe all of the light in the scene by specifying the radiation (or "radiance" along all light rays) arriving at every point in space and from every direction

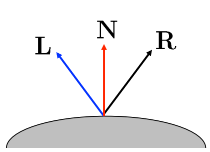


$$R(X, Y, Z, \theta, \phi, \lambda, t)$$

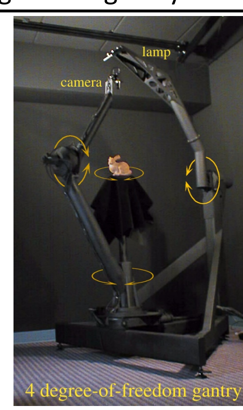
The light field

$$R(X, Y, Z, \theta, \phi, \lambda, t)$$

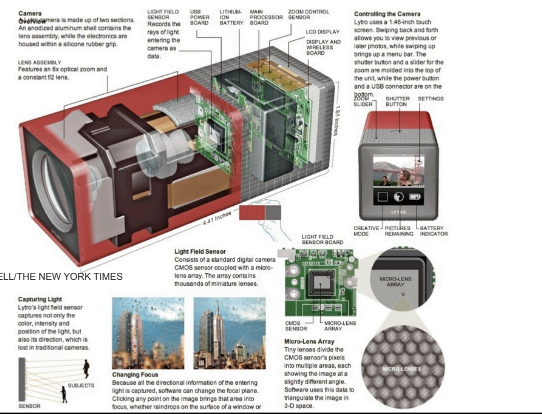
- Known as the **plenoptic function**
- If you know *R*, you can predict how the scene would appear from any viewpoint.
- Common to think of lighting at infinity (a function on the sphere, a 2D space)
- Usually drop λ and time parameters



Stanford light field gantry



Lytro

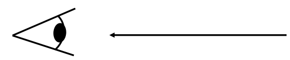

FRANK O'CONNELL/THE NEW YORK TIMES

13

What is light?

Electromagnetic radiation (EMR) moving along rays in space

- R(λ) is EMR, measured in units of power (watts)
  - λ is wavelength

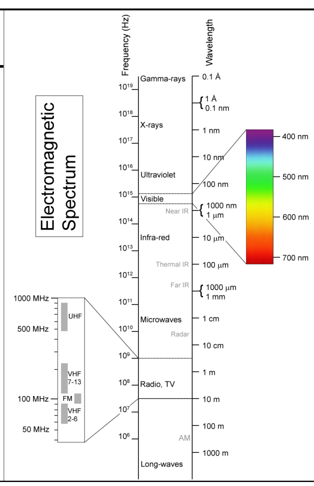


Perceiving light

- How do we convert radiation into “color”?
- What part of the spectrum do we see?

Visible light

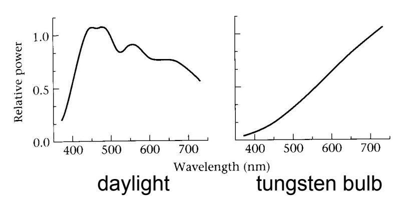
We “see” electromagnetic radiation in a range of wavelengths



Light spectrum

The appearance of light depends on its power **spectrum**

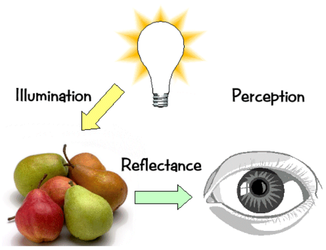
- How much power (or energy) at each wavelength



Our visual system converts a light spectrum into “color”

- This is a rather complex transformation

Light transport



Light sources

Basic types

- point source
- Distant point source
- area source
  - a union of point sources

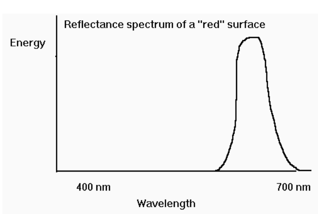
More generally

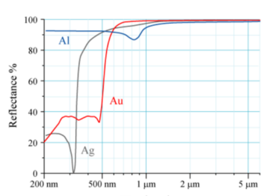
- a light field can describe *any* distribution of light sources

What happens when light hits an object?

Reflectance spectrum (albedo)

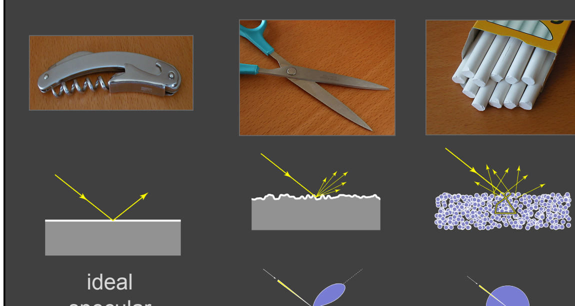
To a first approximation, surfaces absorb some wavelengths of light and reflect others





These spectra are multiplied by the spectra of the incoming light

Typical Reflections



from Steve Marschner

What happens when a light ray hits an object?

Some of the light gets absorbed
- converted to other forms of energy (e.g., heat)

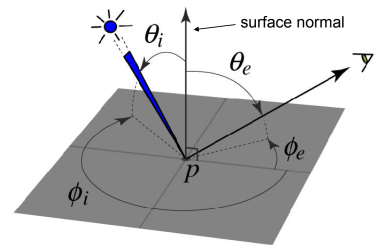
Some gets transmitted through the object
- possibly bent, through "refraction"
- a transmitted ray could possible bounce back

Some gets reflected
- as we saw before, it could be reflected in multiple directions (possibly all directions) at once

Let's consider the case of reflection in detail

The BRDF

The Bidirectional Reflection Distribution Function
- Given an incoming ray (θ_i, ϕ_i) and outgoing ray (θ_e, ϕ_e) what proportion of the incoming light is reflected along outgoing ray?

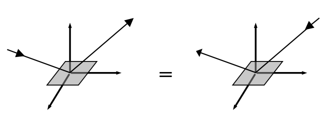


Answer given by the BRDF: $\rho(\theta_i, \phi_i, \theta_e, \phi_e)$

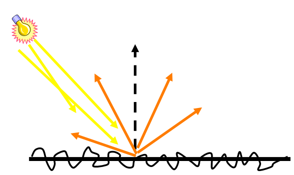
Constraints on the BRDF

Energy conservation
- Quantity of outgoing light ≤ quantity of incident light
  - integral of BRDF ≤ 1

Helmholtz reciprocity
- reversing the path of light produces the same reflectance



Diffuse (Lambertian) reflection

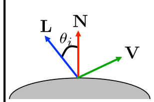


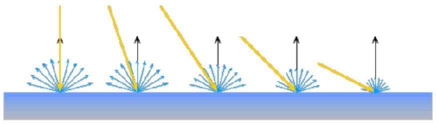
Diffuse reflection
- Dull, matte surfaces like chalk or latex paint
- Microfacets scatter incoming light randomly
- Effect is that light is reflected equally in all directions

Diffuse reflection

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

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





L, **N**, **V** unit vectors
I_e = outgoing radiance
I_i = incoming radiance

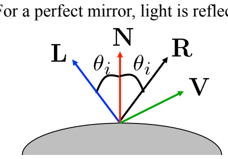
Lambert's Law: $I_e = k_d \mathbf{N} \cdot \mathbf{L} I_i$
k_d is called **albedo**

BRDF for **Lambertian surface**

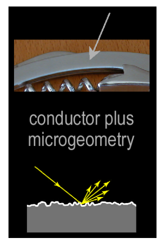
$$\rho(\theta_i, \phi_i, \theta_e, \phi_e) = k_d cos\theta_i$$

Specular reflection

For a perfect mirror, light is reflected about **N**



$$I_e = \begin{cases} I_i & \text{if } \mathbf{V} = \mathbf{R} \\ 0 & \text{otherwise} \end{cases}$$

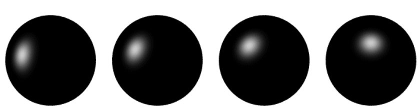


Near-perfect mirrors have a **highlight** around **R**

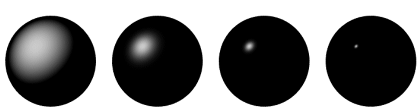
- common model:

$$I_e = k_s (\mathbf{V} \cdot \mathbf{R})^{n_s} I_i$$

Specular reflection



Moving the light source



Changing n_s

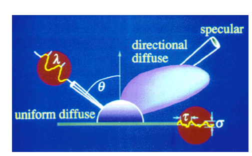
Phong illumination model

Phong approximation of surface reflectance

- Assume reflectance is modeled by three components
  - Diffuse term
  - Specular term
  - Ambient term (to compensate for inter-reflected light)

$$I_e = k_a I_a + I_i \left[k_d (\mathbf{N} \cdot \mathbf{L})_+ + k_s (\mathbf{V} \cdot \mathbf{R})_+^{n_s} \right]$$

L, **N**, **V** unit vectors
I_e = outgoing radiance
I_i = incoming radiance
I_a = ambient light
k_a = ambient light reflectance factor
$(x)_+$ = max(x, 0)



BRDF models

Phenomenological
- Phong [75]
- Ward [92]
- Lafortune et al. [97]
- Ashikhmin et al. [00]

Physical
- Cook-Torrance [81]
- Dichromatic [Shafer 85]
- He et al. [91]

Here we're listing only some well-known examples

Lambertian reflection

$$I_e = k_d \mathbf{N} \cdot \mathbf{L} I_i$$

I_i is light source intensity

Lets assume that $I_i = 1$

can achieve this by dividing each pixel in the image by I_i

$$I = k_d \mathbf{N} \cdot \mathbf{L}$$

- I: image intensity at a single point
- k_d: **Albedo** at a point
- $\mathbf{N}$: Surface **normal** at a point
- $\mathbf{L}$: **Lighting** direction (same for all points)

Shape from shading

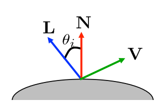
Input:
- Single Image

Output:
- 3D shape of the object in the image

Problem is ill-posed: many shapes can give rise to same image.

Common assumptions:
- Lighting is known
- Lambertian reflectance + uniform albedo
- Boundary conditions are known

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

  $I(x, y) = \cos\theta = 0.5$

  $\theta = 60^\circ, \phi = ?$
- But can be if you add some additional info, for example
  - Assume normals along the sihouette are known
  - Constraints on neighboring normals—"integrability"
  - Smoothness

Surface Normal

$$N = (n_x, n_y, n_z)^T$$

A surface $z(x,y)$

A point on the surface: $(x, y, z(x,y))^T$

Tangent directions

$$t_x = (1,0,z_x)^T \qquad t_y = (0,1,z_y)^T$$

$$N = \frac{t_x \times t_y}{\| t_x \times t_y \|} = \frac{1}{\sqrt{z_x^2 + z_y^2 + 1}} \left(-z_x, -z_y, 1\right)^T$$

Shape from shading

$$I(x,y) = N \bullet L = \frac{-l_1 z_x - l_2 z_y + l_3}{\sqrt{z_x^2 + z_y^2 + 1}}$$

Assume that $L = (0,0,1)^T$

And get that $I(x,y) = N \bullet L = \frac{1}{\sqrt{z_x^2 + z_y^2 + 1}}$

Two unknowns $z_x \quad z_y$

Shape from shading

$$I(x,y) = N \bullet L = \frac{1}{\sqrt{z_x^2 + z_y^2 + 1}}$$

But both unknowns come from an integrable surface: Z(x,y) thus we can use the **integrability** constraint:

$$z_{xy} = z_{yx}$$

Shape from shading

$$I(x,y) = N \bullet L = \frac{1}{\sqrt{z_x^2 + z_y^2 + 1}}$$

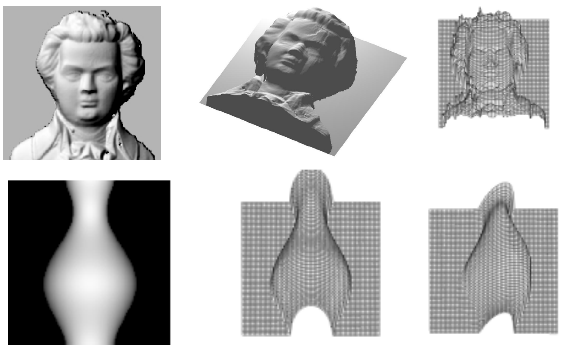
$$\sqrt{z_x^2 + z_y^2 + 1} = \frac{1}{I(x,y)} \Rightarrow \sqrt{z_x^2 + z_y^2} = \sqrt{\frac{1}{I(x,y)^2} - 1}$$

$$\| \nabla z \| = \sqrt{\frac{1}{I(x,y)^2} - 1}$$

is called Eikonal equation
can be solved using variation
of Dijkstra's algorithm

Need to know the extrema
points for this

Results



Shape from shading

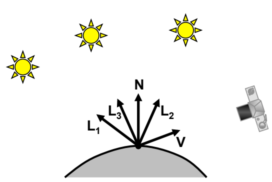
It is hard to get shape from shading work well in practice.

The assumptions are quite restrictive

But this is recovery of **3D** from **single 2D** image

Fewer assumptions are needed if we have several images of the same object under different lightings

Photometric stereo



$$\begin{aligned} I_1 &= k_d \mathbf{N} \cdot \mathbf{L_1} \\ I_2 &= k_d \mathbf{N} \cdot \mathbf{L_2} \\ I_3 &= k_d \mathbf{N} \cdot \mathbf{L_3} \end{aligned}$$

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

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

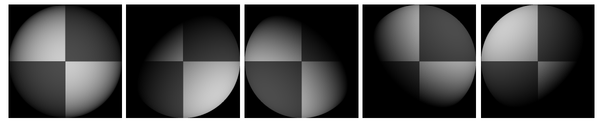
Least squares solution:

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

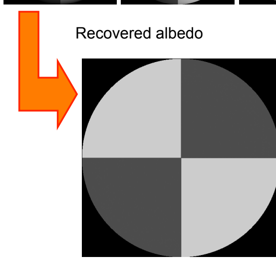
Solve for N, k_d as before

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

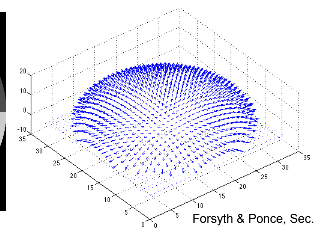
Example



Recovered albedo



Recovered normal field



Forsyth & Ponce, Sec. 5.4

Color images

The case of RGB images

- get three sets of equations, one per color channel:

$$\begin{aligned} \mathbf{I}_R &= k_{dR}\ \mathbf{LN} \\ \mathbf{I}_G &= k_{dG}\ \mathbf{LN} \\ \mathbf{I}_B &= k_{dB}\ \mathbf{LN} \end{aligned}$$

- Simple solution: first solve for **N** using one channel or grayscale
- Then substitute known **N** into above equations to get k_d s

$$k_d = \frac{\sum_i I_i L_i N^T}{\sum_i (L_i N^T)^2}$$

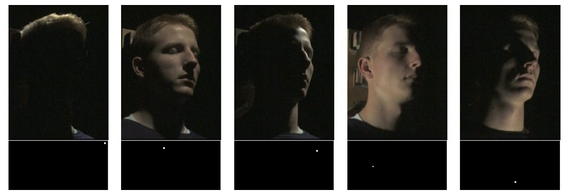
Where do we get the lighting directions?

Capture lighting variation

Illuminate subject from many incident directions

From Ravi Ramamoorthi

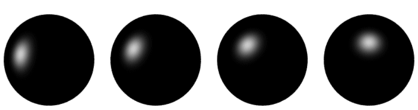
Example images:



From Ravi Ramamoorthi

Computing light source directions

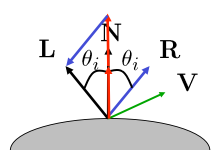
Trick: place a chrome sphere in the scene



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

Recall the rule for specular reflection

For a perfect mirror, light is reflected about N



$$R_e = \begin{cases} R_i & \text{if } \mathbf{V} = \mathbf{R} \\ 0 & \text{otherwise} \end{cases}$$

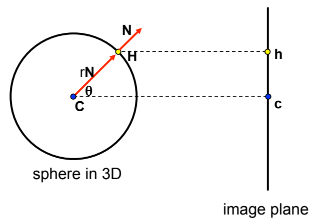
We see a highlight when V = R

- then L is given as follows:

$$\mathbf{L} = 2(\mathbf{N} \cdot \mathbf{R})\mathbf{N} - \mathbf{R}$$

Computing the light source direction

Chrome sphere that has a highlight at position **h** in the image

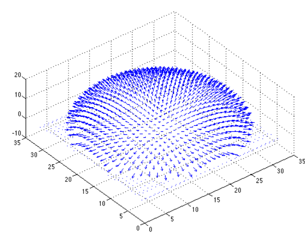


$$N = \begin{pmatrix} \frac{x_h - x_c}{r} \\ \frac{y_h - y_c}{r} \\ \frac{\sqrt{r^2 - x^2 - y^2}}{r} \end{pmatrix}$$

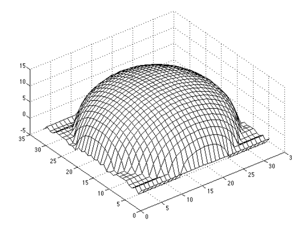
Can compute θ (and hence **N**) from this figure

Now just reflect **V** about **N** to obtain **L**

Depth from normals



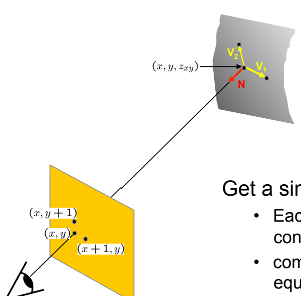
What we have



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
- On the boundary we have only one constraint.

Trick for handling shadows

Weight each equation by the pixel brightness:

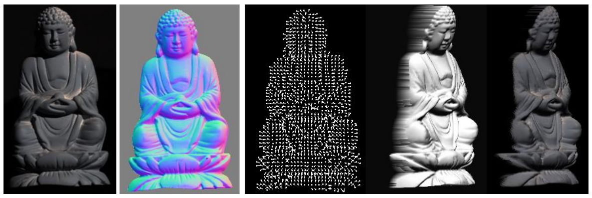
$$I_i(I_i) = I_i[k_d \mathbf{N} \cdot \mathbf{L_i}]$$

Gives weighted least-squares matrix equation:

$$\begin{bmatrix} I_1^2 & \dots & I_n^2 \end{bmatrix} = k_d \mathbf{N}^T \begin{bmatrix} I_1 \mathbf{L_1} & \dots & I_n \mathbf{L_n} \end{bmatrix}$$

Solve for N, k_d as before

Example



Results…

from Athos Georghiades
http://cvc.yale.edu/people/Athos.html

Limitations

- doesn't work for shiny things, semi-translucent things
- shadows, inter-reflections are difficult
- Single light source illumination
- camera and lights have to be distant
- calibration requirements
  - measure light source directions, intensities
  - camera response function

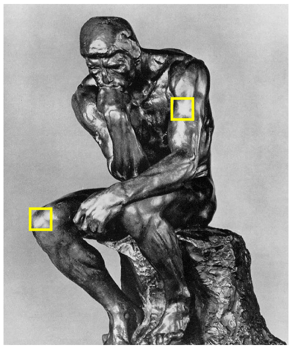
Newer work addresses some of these issues

Some pointers for further reading:

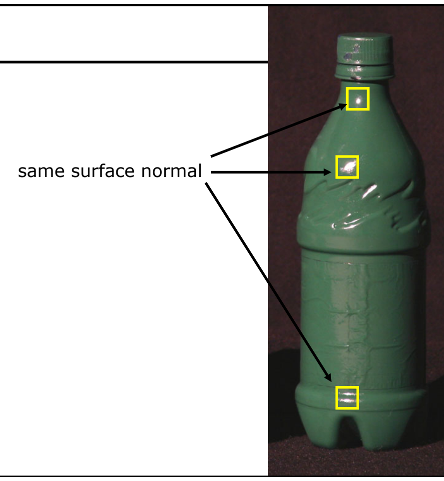
- Zickler, Belhumeur, and Kriegman, "*Helmholtz Stereopsis: Exploiting Reciprocity for Surface Reconstruction.*" IJCV, Vol. 49 No. 2/3, pp 215-227.
- Hertzmann & Seitz, *Example-Based Photometric Stereo: Shape Reconstruction with General, Varying BRDFs*." IEEE Trans. PAMI 2005
- Basri, Jacobs and Kemelmacher Photometric Stereo with General Unknown Lighting, International Journal of Computer Vision (IJCV) 2007

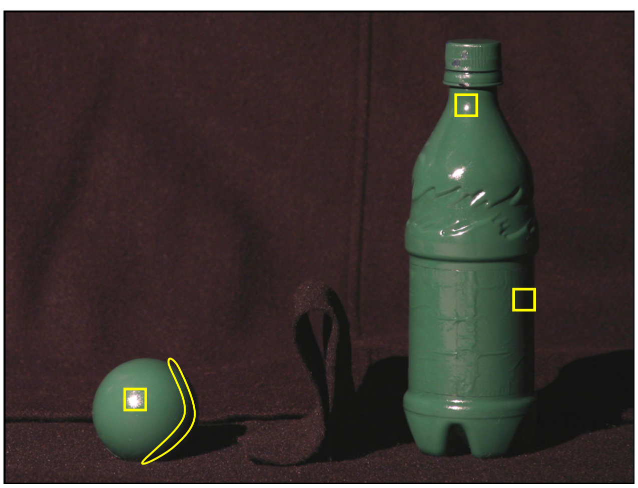
Hertzmann & Seitz, *Example-Based Photometric Stereo: Shape Reconstruction with General, Varying BRDFs*." IEEE Trans. PAMI 2005

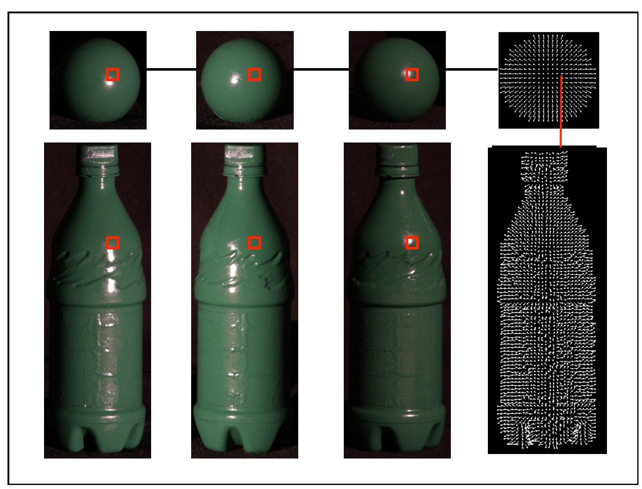
Shiny things



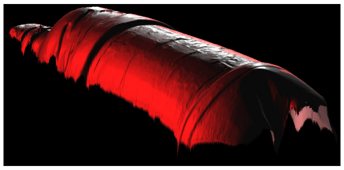
"Orientation consistency"



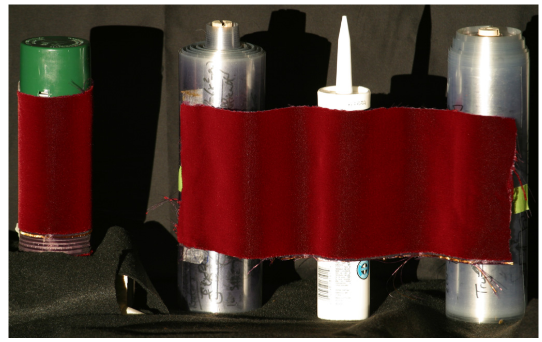




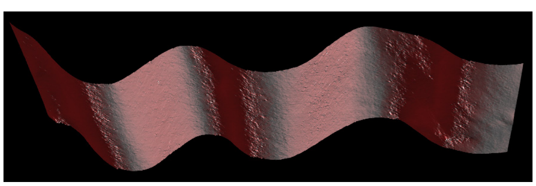
Virtual views

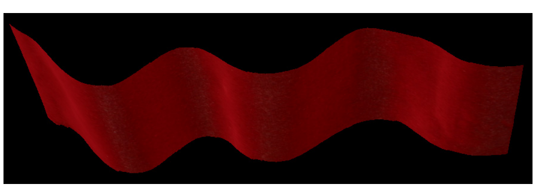


Velvet

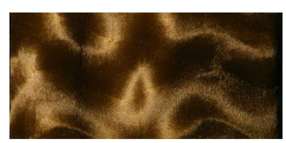


Virtual Views

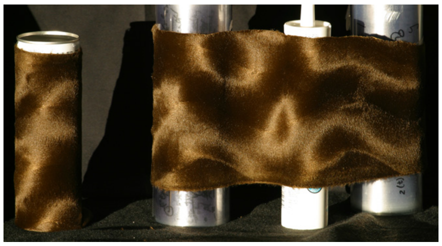




Brushed Fur



Brushed Fur



Virtual Views