Announcements

- Project 2 due today
- Project 3 out today
  - demo session at the end of class

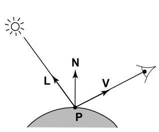
Photometric Stereo

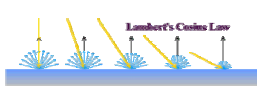
Merle Norman Cosmetics, Los Angeles

Readings

- Forsyth and Ponce, section 5.4

Diffuse reflection





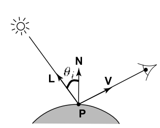
$$R_e = k_d \mathbf{N} \cdot \mathbf{L} R_i$$

image intensity of **P** → $I = k_d \mathbf{N} \cdot \mathbf{L}$

Simplifying assumptions

- $I = R_e$: camera response function f is the identity function:
  - can always achieve this in practice by solving for f and applying f^{-1} to each pixel in the image
- $R_i = 1$: light source intensity is 1
  - can achieve this by dividing each pixel in the image by R_i

Shape from shading



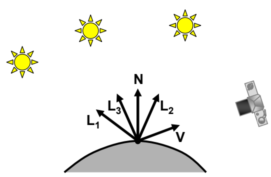
Suppose $k_d = 1$

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

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



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

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:

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

Solve for N, k_d as before

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

Color images

The case of RGB images

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

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

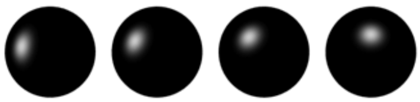
call this **J**

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

$$\begin{aligned} \mathbf{I}_R &= k_{dR}\mathbf{J} \\ \mathbf{J}\cdot\mathbf{I}_R &= k_{dR}\mathbf{J}\cdot\mathbf{J} \\ k_{dR} &= \frac{\mathbf{J}\cdot\mathbf{I}_R}{\mathbf{J}\cdot\mathbf{J}} \end{aligned}$$

Computing light source directions

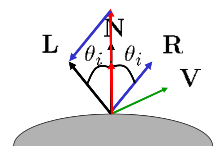
Trick: place a chrome sphere in the scene



- 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**



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

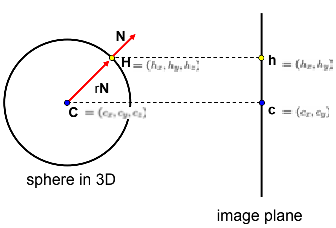
We see a highlight when **V** = **R**

- then L is given as follows:

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

Computing the light source direction

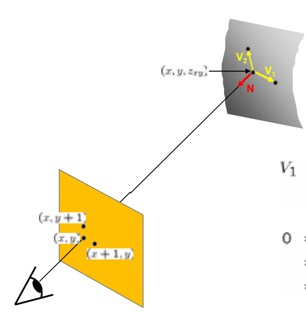
Chrome sphere that has a highlight at position **h** in the image



Can compute **N** by studying this figure

- Hints:
  - use this equation: $\|H - C\| = r$
  - can measure **c**, **h**, and r in the image

Depth from normals



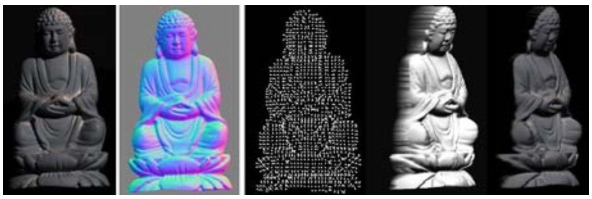
$$\begin{aligned} V_1 &= (x+1, y, z_{x+1,y}) - (x, y, z_{xy}) \\ &= (1, 0, z_{x+1,y} - z_{xy}) \end{aligned}$$

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

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 (project 3)

Project 3



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

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:

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

Solve for N, k_d as before