CS 217: Light and Geometry in Vision **Spring 2009**

Lecture 8 — April 15

Scribe: Tony Tran *Lecturer: Deva Ramanan*

Note: These lecture notes are still rough, and have only have been mildly proofread.

This is the danger environment.

Agenda:

- Light Sources.
- Photometric Stereo - a way to reconstruct 3D surfaces given multiple images.

8.1 Light Sources

We previously derived the appearance of diffuse objects under a far away point light source as:

$$L_e(x, \theta_e, \phi_e) = \frac{1}{\pi}\rho_d(x)cos(\theta_i)$$

where θ_i = angle between surface normal and point source.

8.1.1 Near-by spherical light source

Consider a near-by spherical light source (e.g., flash photography) as depicted below:

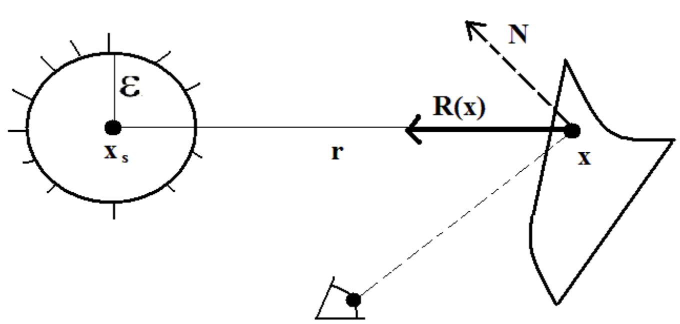


Figure 8.1. Near by spherical light source with radius ϵ and a lambertian surface

Assume the radiance leaving the point source is constant (E) across all directions:

$$L(\alpha_s, \theta_e, \phi_e) = E \tag{8.1}$$

$$L_e(x, \theta_e, \phi_e) = L_e(x) = \int_{2\pi} \frac{1}{\pi} \rho_d(x) L_i(x, \theta_i, \phi_i) cos(\theta_i) dw_i \tag{8.2}$$

$$dw_i = \frac{cos(\tilde{\theta}) dA}{R^2} = \frac{\pi \varepsilon^2}{\|r(x)^2\|}, r(x) = x_s - x \tag{8.3}$$

$$= \frac{1}{\pi} \rho_d(x) E cos(\theta_i) \frac{\pi \varepsilon^2}{\|r(x)^2\|} \tag{8.4}$$

$$= \rho_d(x) \varepsilon^2 E N(x)^T r(x) \tag{8.5}$$

8.1.2 Special case

If we assume:

$$r(x) = r_o + Ar(x)$$

$$r_o >> \max_x |(\Delta r(x))|$$

$$r_o = avg(x_s - x)$$

.

Then we can do the following: Let $S = \varepsilon^2 E \frac{r_o}{\|r_o\|^3}$. This will result in:

$$L_e = \rho_d(x) N(x)^T S$$

8.2 Shadows (Qualitative descriptions)

8.2.1 Point Source

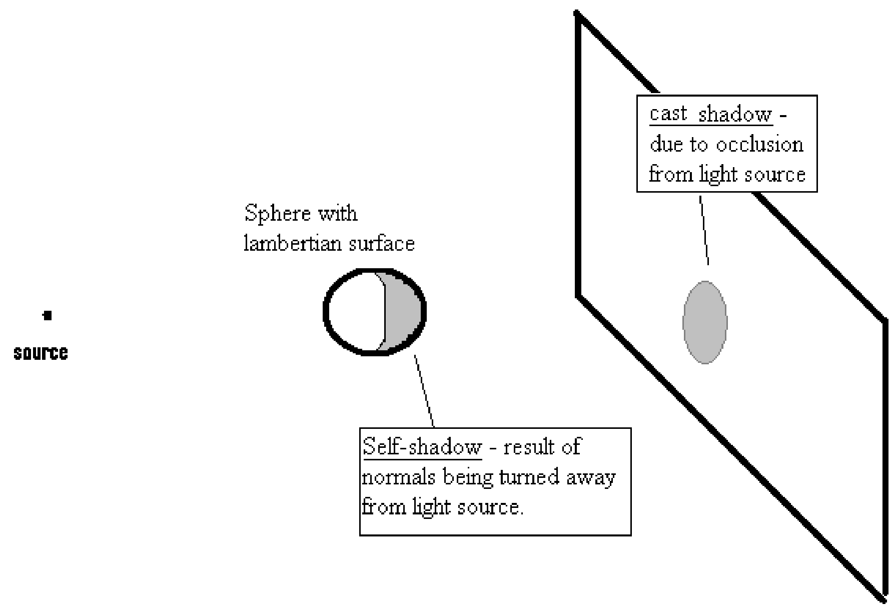


Figure 8.2. Shadows resulting from a point light source

8.2.2 Spherical Source

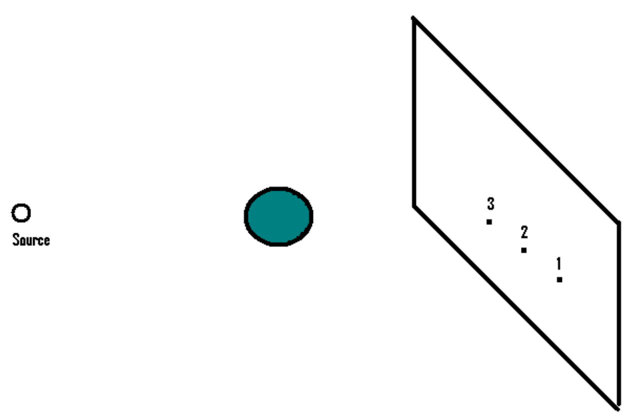


Figure 8.3. Spherical light source

Question: What will the incoming hemisphere at point 1, 2, and 3 look like?

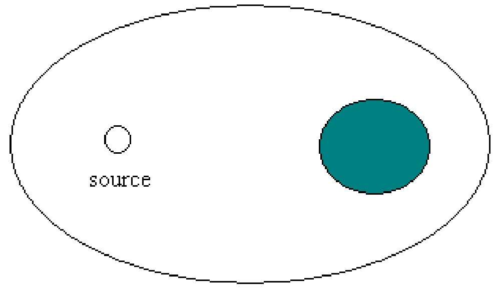
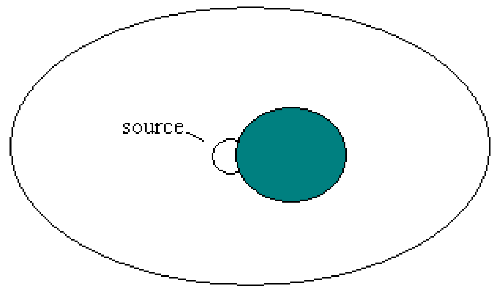
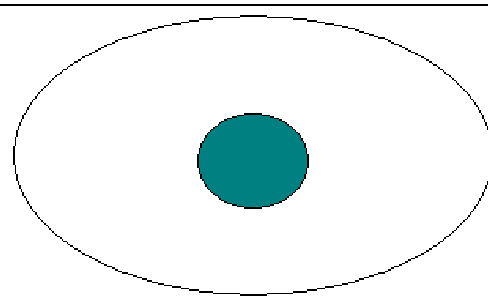
location	Description	Incoming hemisphere
1	Points out of shadow	
2	Points in the **penumbra** of shadow	
3	Points in the **unbra** of shadow.	

Figure 8.4. This table depicts the incoming hemisphere at points 1, 2, and 3.

8.3 Photometric stereo

8.3.1 Goal

The goal of Photometric stereo is to reconstruct object shape by estimating the normals (N(x,y)) and albedo values (P(x,y)) for every (x,y) location.

8.3.2 Setup

The following assumptions and notations are used for performing photometric stereo:

- The object has a lambertian surface with spatially varying albedo.
- The spherical source is far away from the object.

- We are given N-images of the object under known lighting $S_1, ..., S_n$. Note that S_i is simply a vector that points in the direction of the source.
- The 3D points of the object can be represented as:

$$P_i = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

- Orthogrpahic projection to image pixels (x_i, y_i), with $0 < x <$(width of image) and $0 < y <$(height of image)
- $I(x, y) = L_e(x, y) = g(x, y)^T S$, where $g(x, y) = \rho_d(x, y)N(x, y)$. The g term can also be thought of as the normal at (x,y,z) scaled by the albedo at (x,y,z).

8.3.3 Estimating Normals and Albedos

Assuming that we have N-images, we will will be given:

$$I_1(x, y)...I_N(x, y)$$

$$S_1...S_N, S_i \in \mathbb{R}^3 (3x1) vector$$

We can then define:

$$V = \begin{pmatrix} S_1^T \\ ... \\ S_N T \end{pmatrix}$$

$$C(x, y) = \begin{pmatrix} I_1(x, y) \\ ... \\ I_N(x, y) \end{pmatrix}$$

In a noise-free case, we get:

$$C(x, y) = V g(x, y)$$

A few things we can note about this noise-free case are:

- We will have a linear system for each pixel
- Can solve (for g) by inverting V
- We will have 3 uknown g(x,y)
- We NEED $N \geq 3$.

The right way of handling this when we have more than 3 light sources is to find g such that:

$$g = \min_{g} \|C(x, y) - Vg(x, y)\|^2$$

Note that we can solve this with least squares: $g(x, y) = (V^T V)^{-1} V^T C(x, y)$.

We can now estimate the albedo and normal as:
Albedo: $p(x, y) = \|g(x, y)\|$
Normal: $N(x, y) = \frac{g(x,y)}{p(x,y)}$

At this point, we can use some shape-from-normal algorithm to reconstruct the object.