Announcements

- Project 3a due today
- Project 3b due next Friday

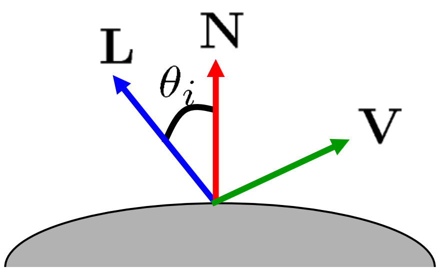
Last time we learned:

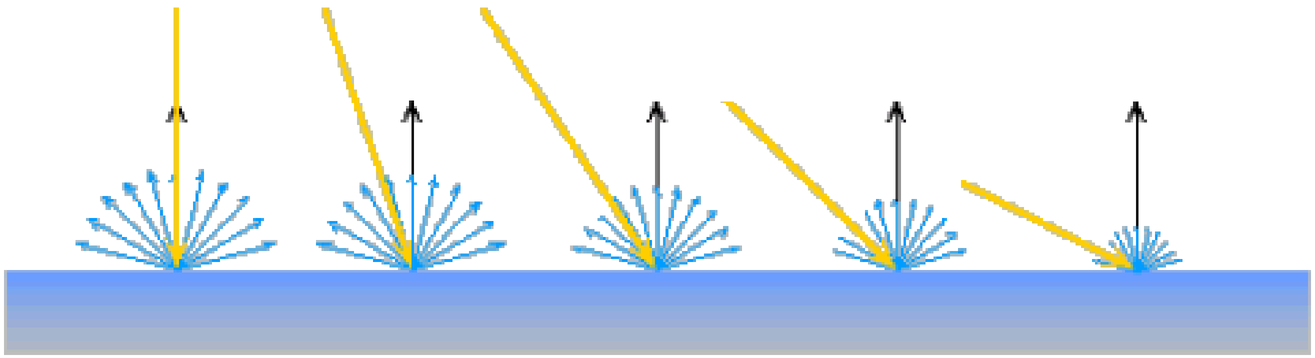
- Light, light field
- BRDF
- Reflectance models:
    - Specular
    - Lambertian
    - Phong
- etc.

Lambertian (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**

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

Lambertian reflection

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

is light source intensity

Lets assume that $I_i = 1$

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

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

image intensity at a single point

Lighting direction (same for all points)

Albedo at a point

Surface **normal** at a point

Shape from shading

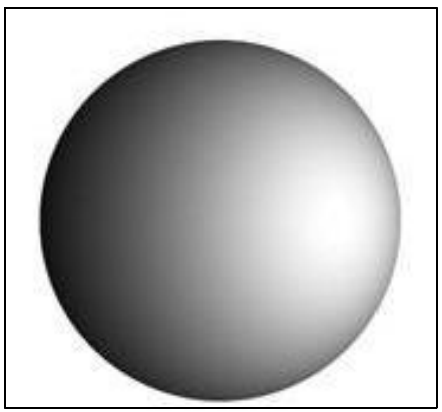
<u>Input:</u>

- Single Image

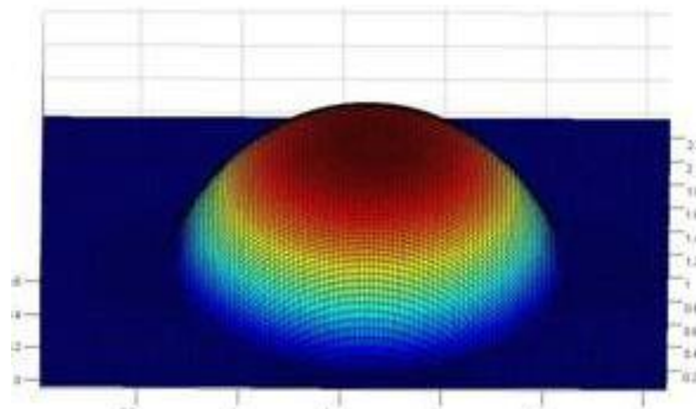
<u>Output:</u>

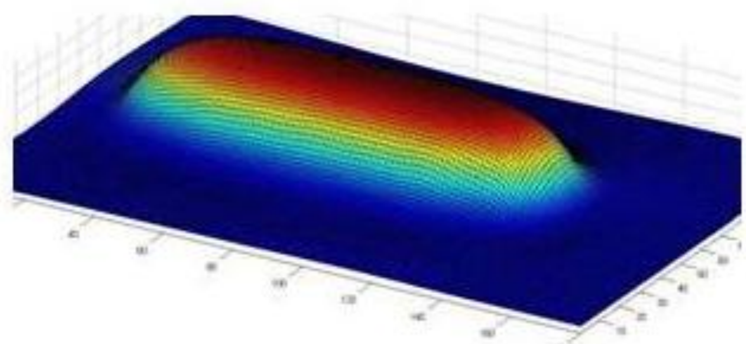
- 3D shape of the object in the image

I(x,y)



Z(x,y)





Shape from shading

Input:

- Single Image

Output:

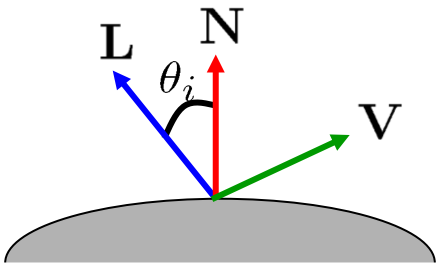
- 3D shape of the object in the image

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

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

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

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

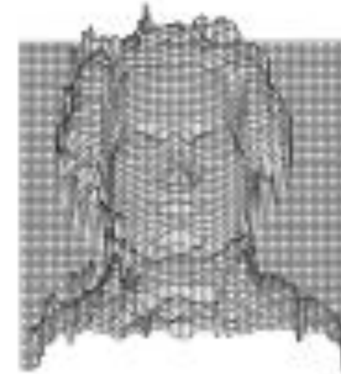
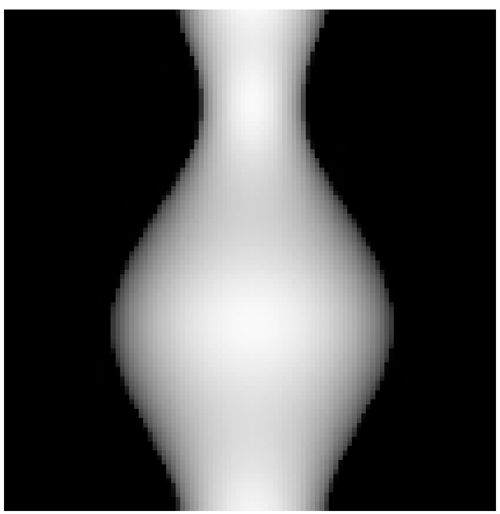
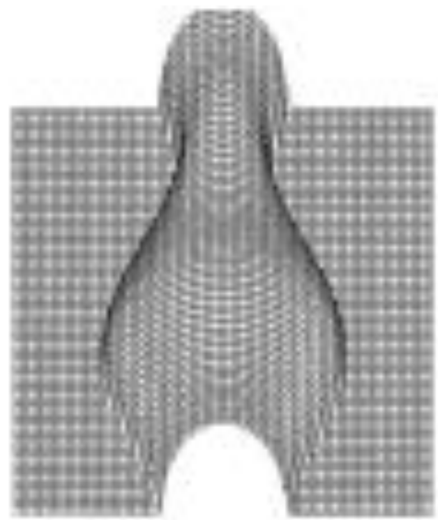
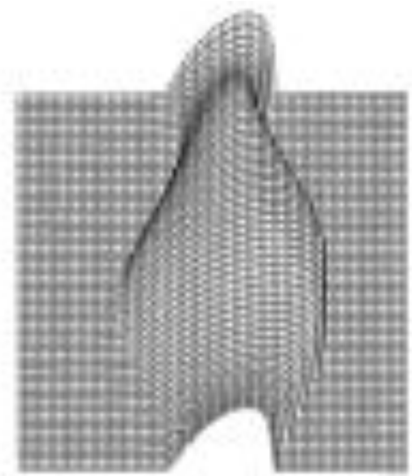
$$\| \nabla z \| = \sqrt{\frac{1}{I(x,y)^2} - 1}$$

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

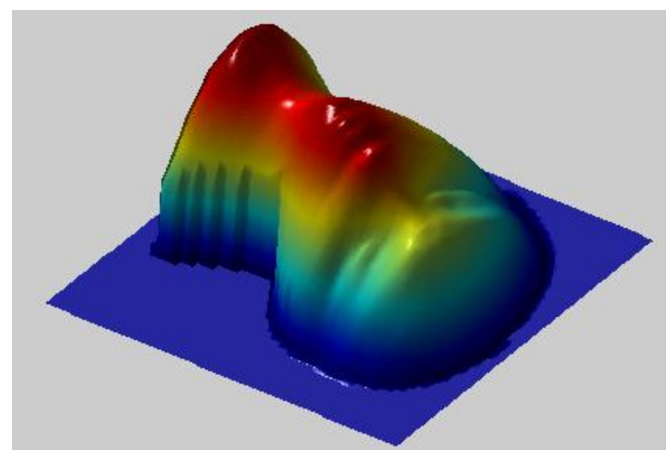
Photometric stereo

<u>Input:</u>

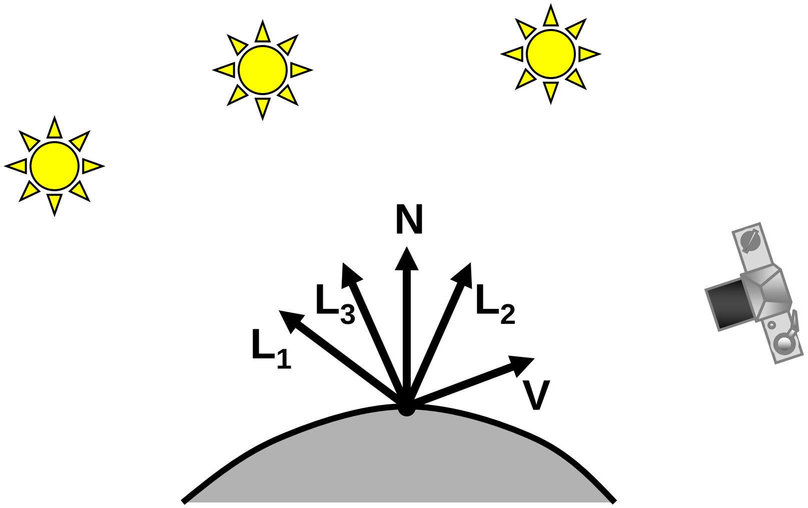
- Several Images
  * same object
  * different lightings
  * same pose

<u>Output:</u>

- 3D shape of the object in the images
- Albedo
- Lighting



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

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

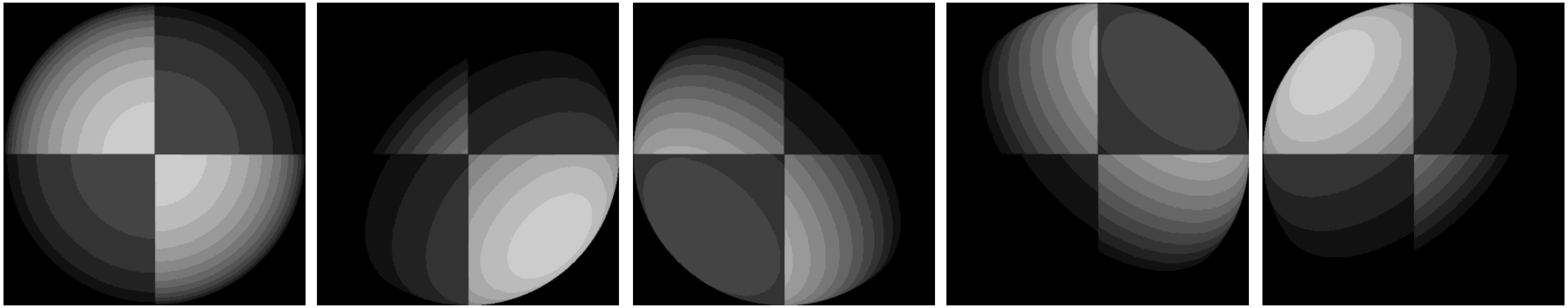
Least squares solution:

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

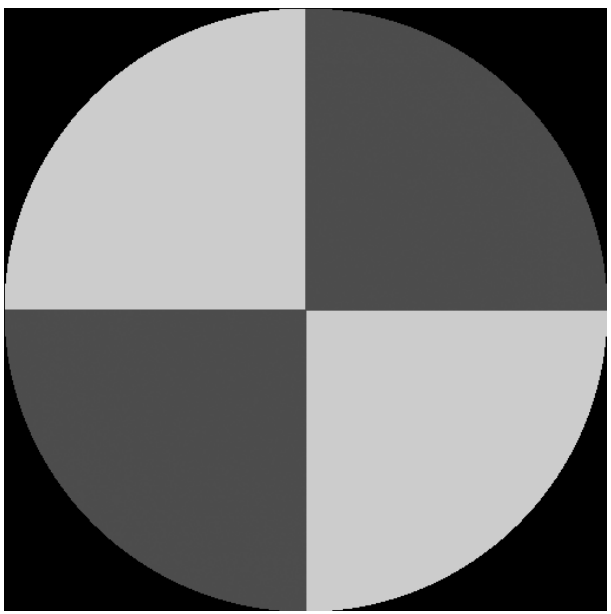
Solve for N, k_d as before

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

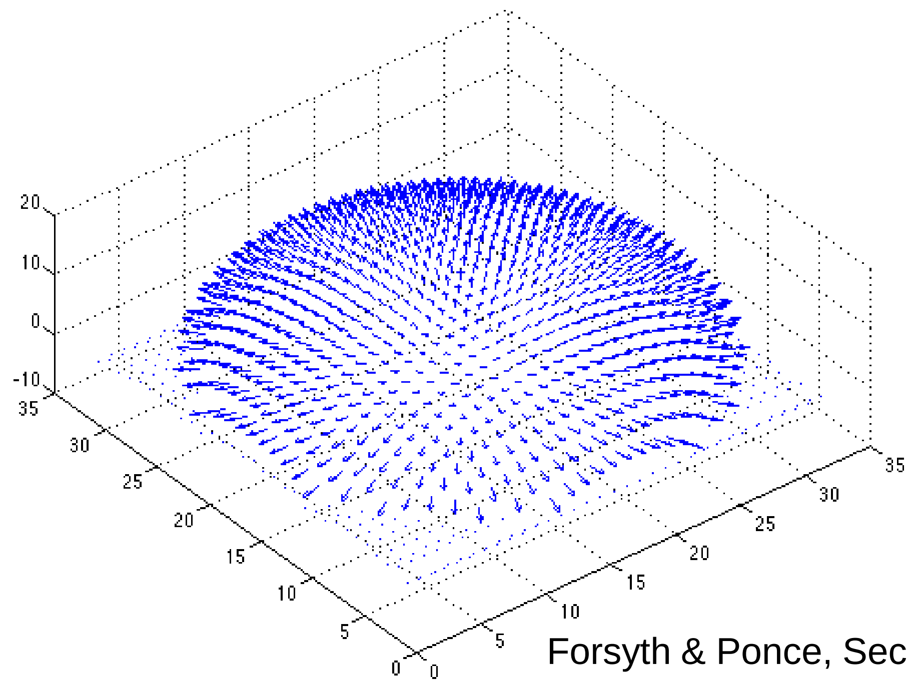
Example



Recovered albedo



Recovered normal field



Forsyth & Ponce, Sec. 5.4