

lecture 16

shape from shading



Given an image $I(x,y)$, compute a 3D model $\{(x,y,z)\}$ that explains the image's intensities.

We need:

- a shading model
- some of the model's parameters

Typical Assumptions:

- weak perspective (see below)
- surface is smooth
- surface is Lambertian
- surface has uniform reflectance
- illumination from single direction

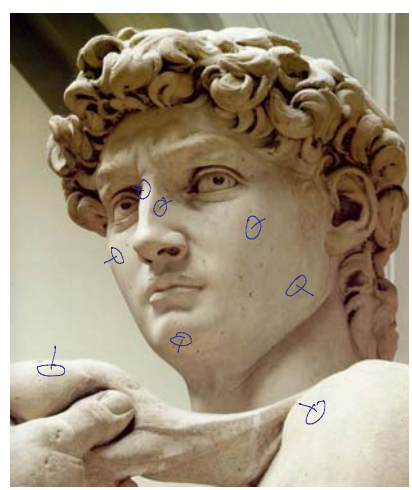
Shape-from-shading on a sunny day

Assumptions:

- illumination from single direction

$$\Rightarrow E(x,y) = \vec{n}(x,y) \cdot \vec{l}_{src}$$

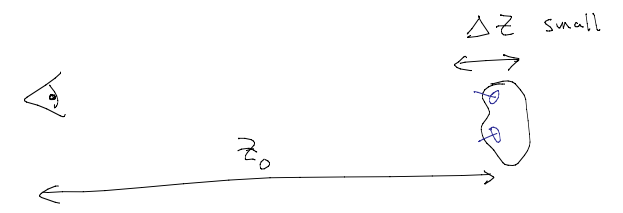
$E(x,y)$ units chosen so that $E=1$ when $\vec{n} = \vec{l}_{src}$



⊙ represents a disk lying tangent to surface, and normal vector $\vec{n}(x,y)$

Goal is to compute $\vec{n}(x,y)$ at each pixel.

Weak Perspective ($z \approx z_0$)



Assume

$$x \approx f \frac{X}{z_0}$$

$$y \approx f \frac{Y}{z_0}$$

We are estimating the shape of a **small surface**, not the 3D geometry of the whole scene.

Weak Perspective

$$E(x, y) = E\left(\frac{f}{z_0} x, \frac{f}{z_0} y\right)$$

↑
pixels

Let surface depth map be $z(x, y)$.

$E(x, y)$ depends on $z(x, y)$
via a shading model.

[Notation: I will write $E(X, Y)$
from now on.]

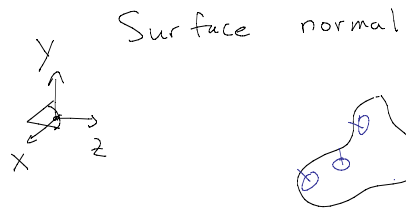
Shape from shading on a sunny day

Recall lecture 6

$$E(x, y) = \vec{n}(x, y) \cdot \vec{l}_{src}$$

↑
surface normal
at $(x, y, z(x, y))$

How is $\vec{n}(x, y)$ determined by $z(x, y)$?



$n(x, y)$ is determined by $(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y})$. How?

Consider two nearby points on the surface
 (X, Y, z) and $(X + \Delta X, Y + \Delta Y, z + \Delta z)$.

$$z(x + \Delta x, y + \Delta y)$$

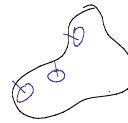
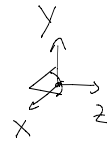
$$\approx z(x, y) + \frac{\partial z}{\partial x} \Delta x + \frac{\partial z}{\partial y} \Delta y$$

$$\Rightarrow \Delta z = \frac{\partial z}{\partial x} \Delta x + \frac{\partial z}{\partial y} \Delta y$$

$$\Rightarrow (\Delta x, \Delta y, \Delta z) \cdot \left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1\right) = 0$$

↑
any step tangent
to surface

↑
∴ this is
the surface
normal
direction



Unit Surface Normal

$$\vec{n}(x, y) = \frac{1}{\sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2 + 1}} \left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1\right)$$

Shape from shading on a sunny day

$$E(x, y) = \vec{n}(x, y) \cdot \vec{l}_{src} = \frac{\left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1\right) \cdot \vec{l}_{src}}{\sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2 + 1}}$$

Note: $E(x_0, y_0)$ and \vec{l}_{src} do not
uniquely determine $\vec{n}(x_0, y_0)$.



Shape from shading on a sunny day

$$E(x, y) = \frac{\left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1\right) \cdot \vec{l}_{src}}{\sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2 + 1}}$$

Given an image patch,

- assume or estimate \vec{l}_{src}
- estimate $\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}$ at each pixel
- & integrate to estimate $z(x, y)$

There have been hundreds of attempts/
methods to solve this problem.

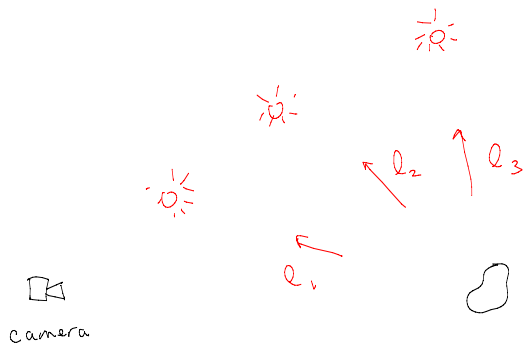
☹ "Exact solutions" are not
useful in practice since the
assumptions are so strong.

- Real surfaces are not exactly Lambertian.
- Single light source model \vec{l}_{src}
rarely holds
- Light source is not known
- Surface reflectance rarely uniform.

SFS - Sunny Day is historically very important.

- Understanding why a problem is
difficult led to many techniques
for solving related problems
- Shading is still an important
visual cue. Just don't
expect an exact solution...

Related Problems: (1) Photometric Stereo



Idea is to use multiple light source directions. $\vec{l}_1, \vec{l}_2, \vec{l}_3$

$$\begin{bmatrix} E_1(x,y) \\ E_2(x,y) \\ E_3(x,y) \end{bmatrix} = \begin{bmatrix} l_x^1 & l_y^1 & l_z^1 \\ l_x^2 & l_y^2 & l_z^2 \\ l_x^3 & l_y^3 & l_z^3 \end{bmatrix} \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}$$

If \vec{l} vectors are known (lab setup) then 3×3 l -matrix can be inverted and we can solve for \vec{n} .

$$\begin{bmatrix} E_1(x,y) \\ E_2(x,y) \\ E_3(x,y) \end{bmatrix} = \begin{bmatrix} l_x^1 & l_y^1 & l_z^1 \\ l_x^2 & l_y^2 & l_z^2 \\ l_x^3 & l_y^3 & l_z^3 \end{bmatrix} \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \cdot \rho(x,y)$$

↑
reflectance

Note: with three \vec{l} 's, we can in principle solve for a scalar reflectance too! (up to an unknown scale).

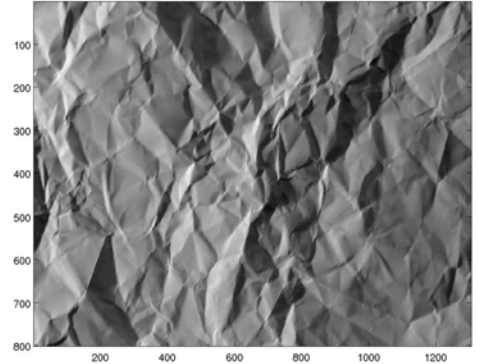
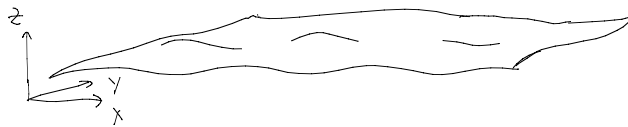


- ☹️ still need to handle shadows, and non-Lambertian effects
- 😊 Using more than 3 images helps!

Related Problems (2): linear SFS

$$n(x,y) = \frac{(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1)}{\sqrt{(\frac{\partial z}{\partial x})^2 + (\frac{\partial z}{\partial y})^2 + 1}}$$

Consider case: $\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}$ are small



$$n(x,y) = \frac{(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1)}{\sqrt{(\frac{\partial z}{\partial x})^2 + (\frac{\partial z}{\partial y})^2 + 1}}$$

Take Taylor series expansion of denominator

$$\frac{1}{\sqrt{(\frac{\partial z}{\partial x})^2 + (\frac{\partial z}{\partial y})^2 + 1}} \approx 1 - \frac{1}{2} \left[\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2 \right] + \text{higher order terms}$$

$$\approx 1, \quad \text{if } \frac{\partial z}{\partial x}, \frac{\partial z}{\partial y} \text{ are small}$$

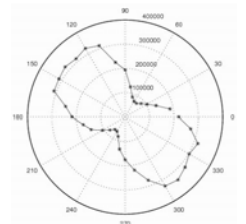
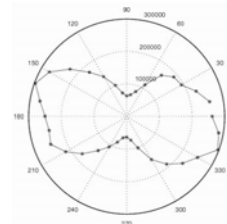
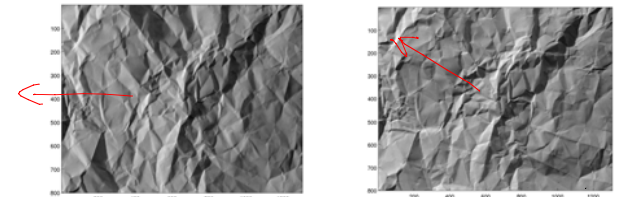
$$E(x,y) = \frac{(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1) \cdot \vec{l}_{src}}{\sqrt{(\frac{\partial z}{\partial x})^2 + (\frac{\partial z}{\partial y})^2 + 1}}$$

$$\approx (\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1) \cdot (l_x, l_y, l_z)$$

linear shading

If $|l_z| \ll |l_x, l_y|$, then we can get significant shading effects!

(l_x, l_y) direction



$$E(x, y) = \left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, -1 \right) \cdot (l_x, l_y, l_z)$$

Solution:

Integrate along lines in direction (l_x, l_y)

eg. if $l_y = 0$, then:

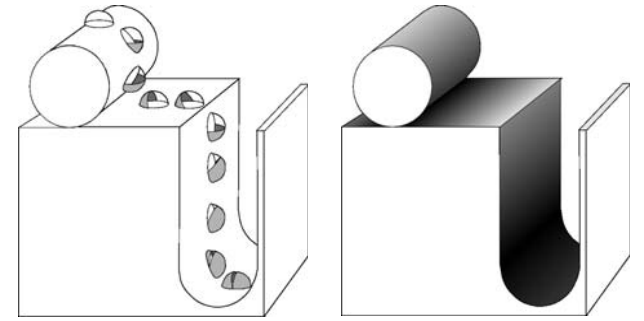
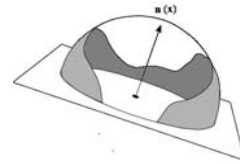
$$\int_{x_{min}}^{x_0} E(x, y) dx = -l_z(x_0 - x_{min}) + l_x \int_{x_{min}}^{x_0} \frac{\partial z}{\partial x} dx$$

data \nearrow $= -l_z(x_0 - x_{min}) + l_x(z(x_0) - z(x_{min}))$

Related Problems (3) Shape from shading on a cloudy day

$$E(x, y) = \int_{\vec{l} \in V(x, y)} \vec{n}(x, y) \cdot \vec{l} d\Omega$$

$V(x, y)$ is the set of directions in which the sky is visible from (x, y, z) .



$$E(x, y) = \frac{1}{2\pi} \int_{\vec{l} \in V(x, y)} \vec{n}(x, y) \cdot \vec{l} d\Omega$$

$$\approx \frac{1}{2\pi} \int_{\vec{l} \in V(x, y)} d\Omega$$

= η_0 of sky that is visible from x, y, z

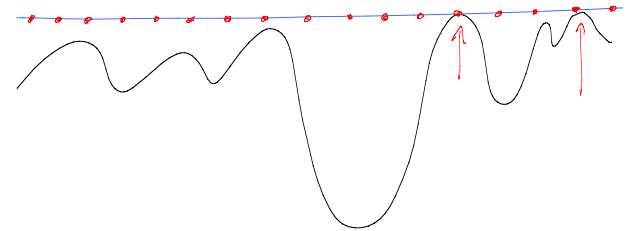
Shape from shading on a cloudy day

As always for SFS, assume weak perspective.

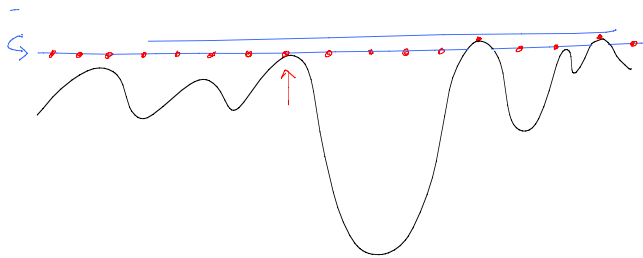
Given an image $E(x, y)$, find a surface depth map $z(x, y)$ that is consistent with

$$E(x, y) = \frac{1}{2\pi} \int_{V(x, y)} d\Omega$$

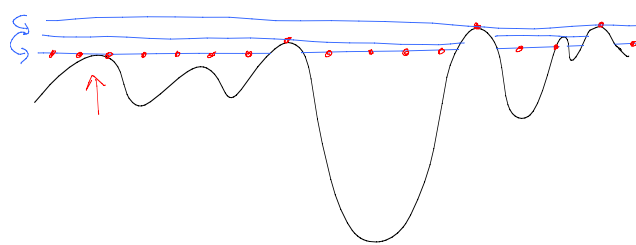
Water Spider Algorithm.



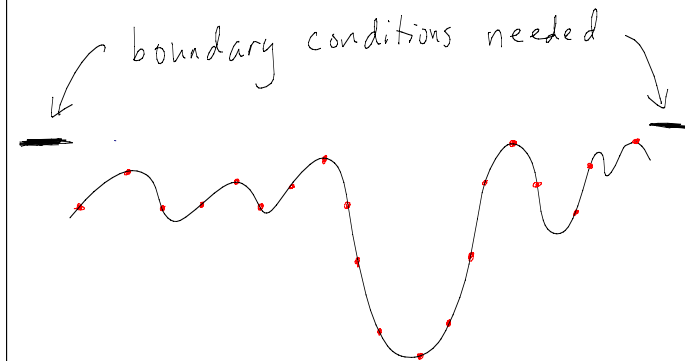
Water Spider Algorithm.



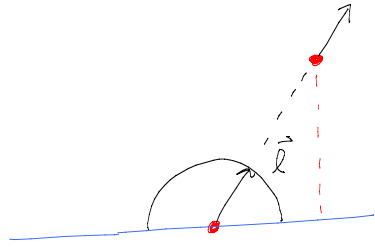
Water Spider Algorithm.



Water Spider Algorithm.



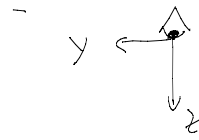
Local visibility constraints



Is the sky visible in direction \vec{l} ?
Ask your neighbor.

Limitations

- $E(x, y) = \frac{1}{2\pi} \int d\Omega$ is a very rough approximation
 $V(x, y)$



More general viewing geometry would be good.



Shape from shading is not a single problem

- sunny day (classical)
- photometric stereo
- linear
- cloudy day