Light and Computer Vision: Shape from Shading

CS 763
Sharat@iitb.ac.in

Computer Vision is an old (new) topic. Slides for these have been taken with permission from various sources and compiled for this edition of cs 763

Contents

What Did We Do Thursday?

- Introduction
- Concept of scene and image irradiance
  - Proving (falsifying) a fundamental assumption
- Problem Definition
  - Qualitative, Quantitative
- Reflectance function
- Shape from shading
- Photometric stereo

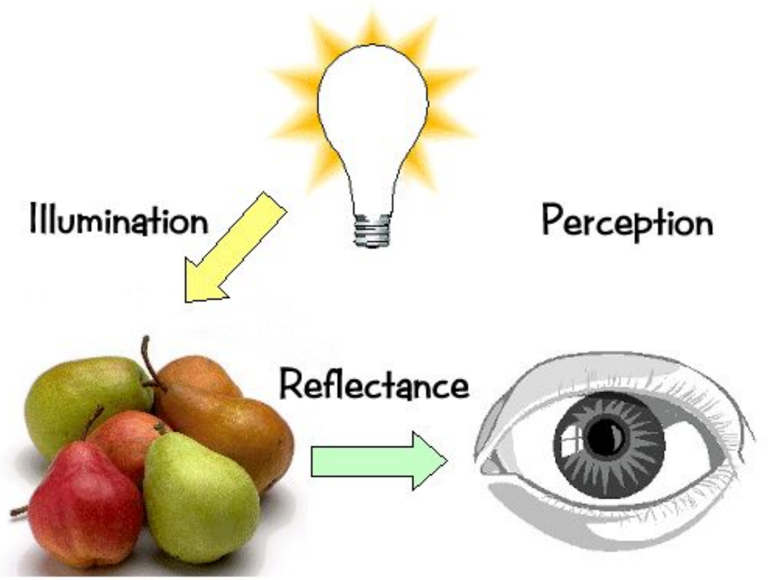
Recap

Agenda: What We Want To Do

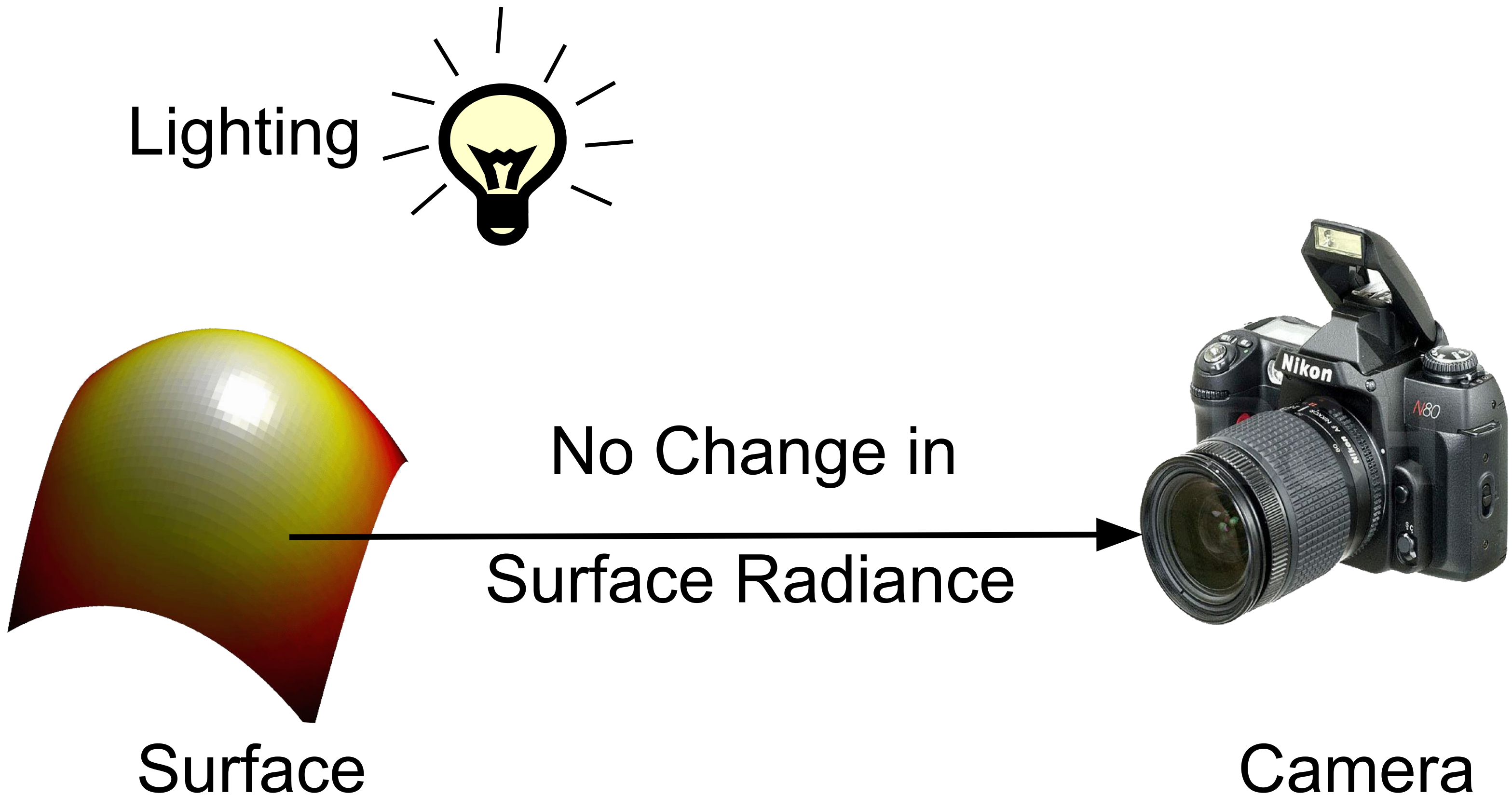
- Want: Reconstruct three dimensional structure from a photo
- Why (consider face as an example)
  - Relighting, generate new view, generate expressions
  - Fundamental problem in vision
- How we will approach this
  - We will take a “first principles approach”
  - Some (possibly) new mathematical concepts

Image Understanding: First Principles

- Able to see things
  - because there is light
- Humans able to interpret
  - perception
- Computer vision
  - hardware (camera)
  - software (intelligence)



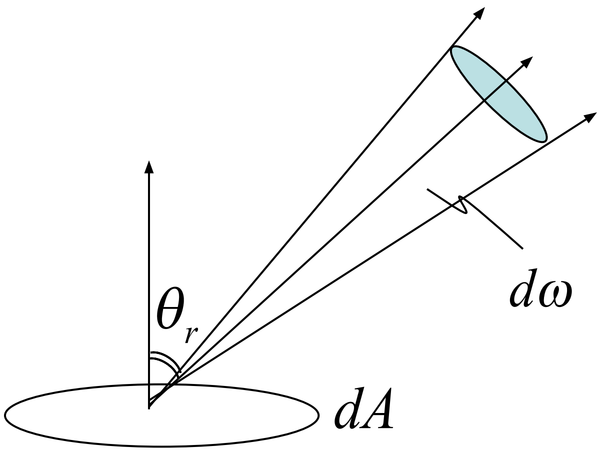
Fundamental Assumption in Vision



- What we capture is what is out there
- We need to define what is “out there”

Radiance

- Radiance is **the** quantity of interest
- Compare definition of [mass]
- Why: Radiance remains conserved in space as it propagates along ray

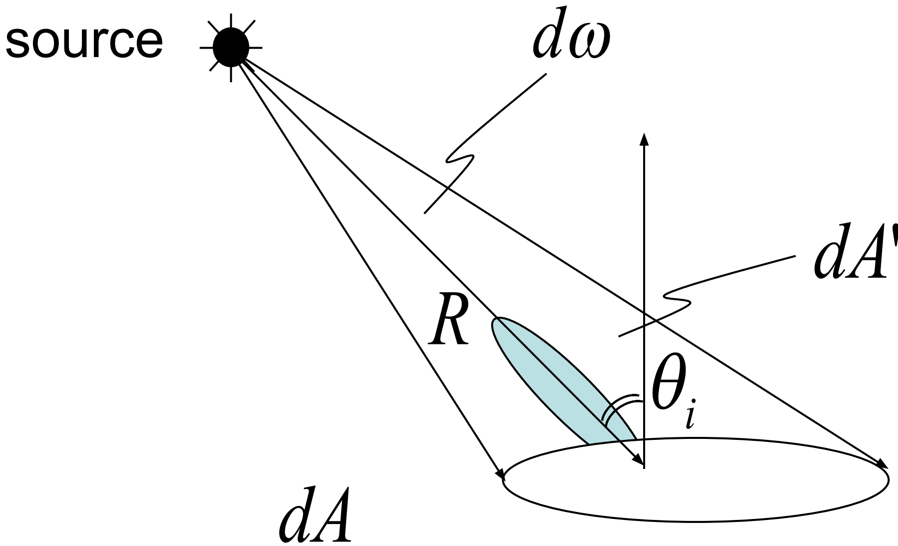


$$L = \frac{d^2\Phi}{(dA \cos\theta_r)\ d\omega}$$

(watts / m^2 steradian)

Image Irradiance

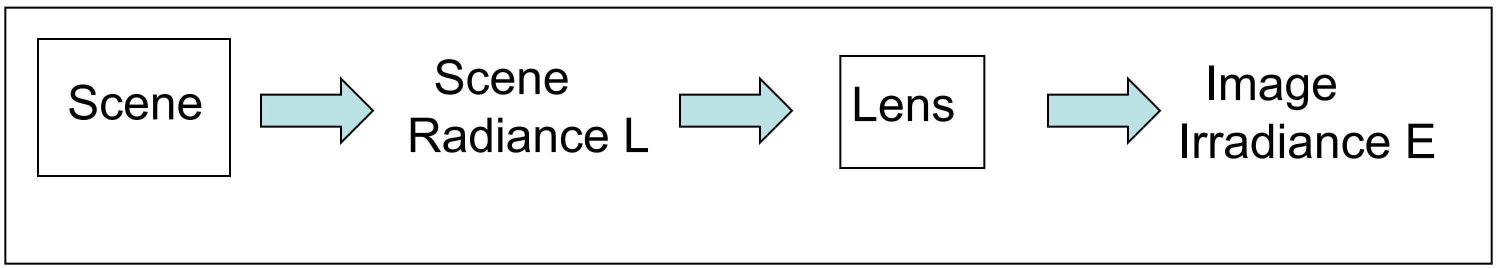
- Irradiance is the power per unit area



$$E = \frac{d\Phi}{dA}$$

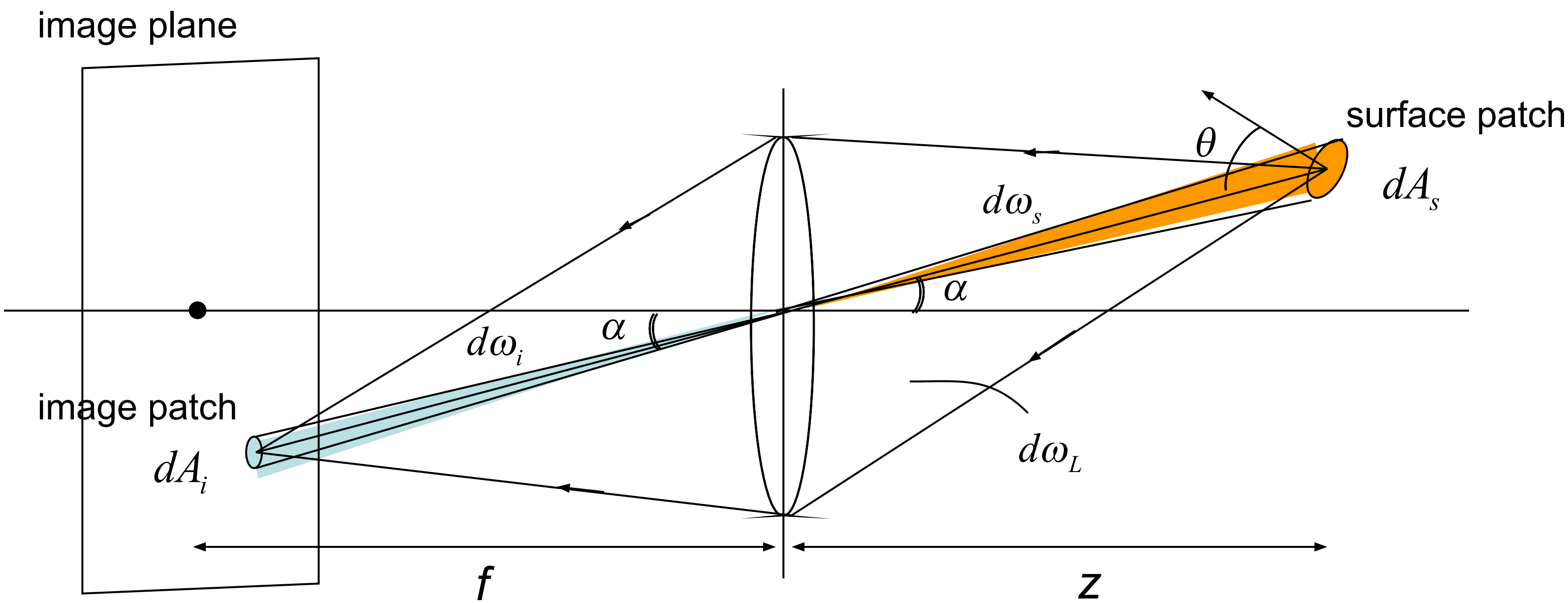
Scene and image: Relation

Before light hits the image plane



Goal: Measured $I(x,y)$ should be "what is out there?" (we don't care for constant factors)

Relation between E and L



$$\boxed{L\,(dA_s\cos\theta)\,d\omega_L = E\,dA_i} \longrightarrow (3)$$

From (1), (2), and (3):

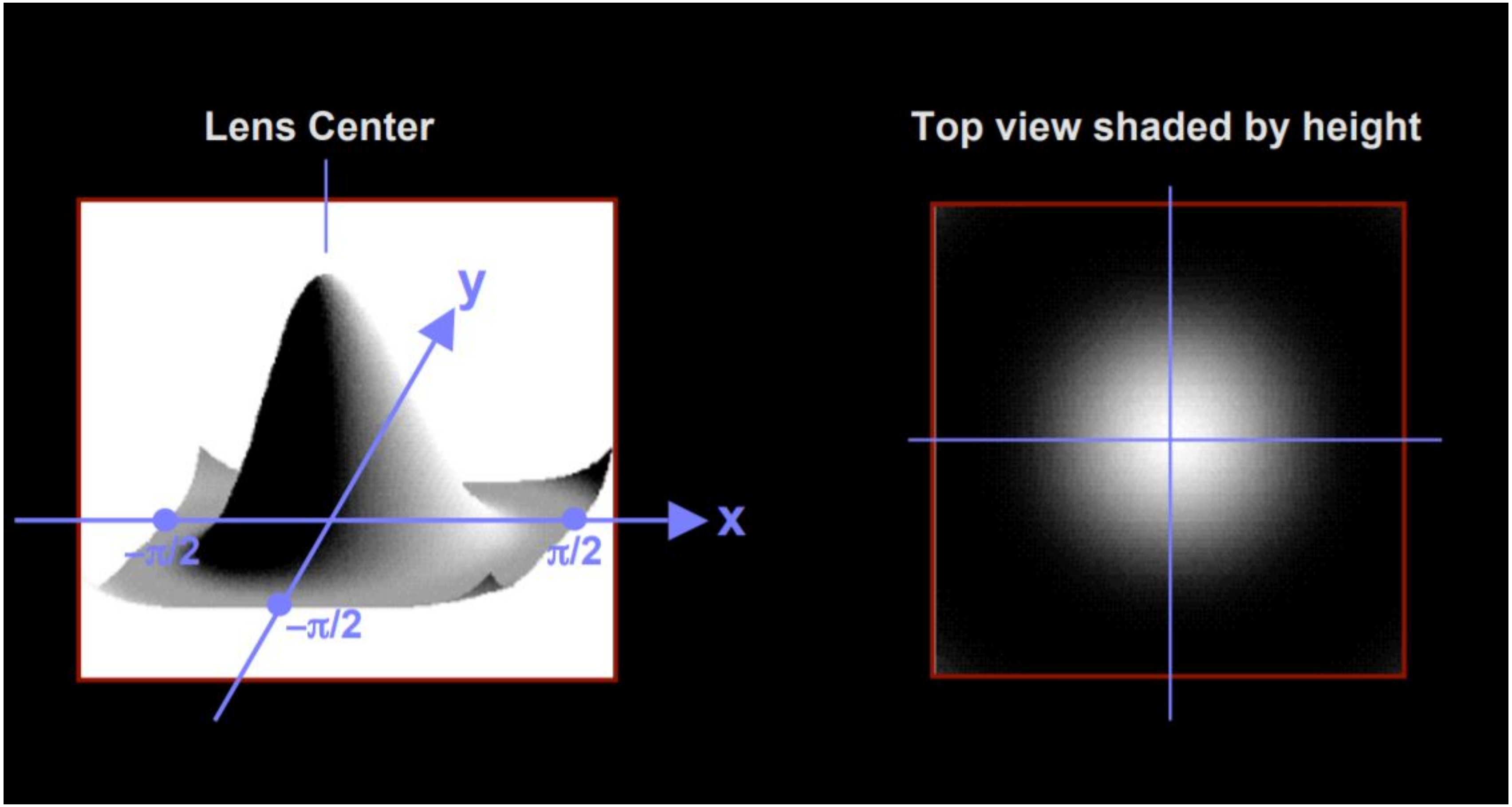
$$\boxed{E = L\,\frac{\pi}{4}\left(\frac{d}{f}\right)^2 \cos\alpha^4}$$

Linear relation?

Contents

- Introduction
- Concept of scene and image irradiance
  - Proving (falsifying) fundamental assumption
- Reflectance function
- Photometric stereo
- Shape from shading

Off axis cut off



Pixel Value & Irradiance: Relation

Before light hits the image plane

Scene → Scene Radiance L → Lens → Image Irradiance E

Linear Mapping!

After light hits the image plane

Image Irradiance E → Camera Electronics → Measured Pixel Values, I

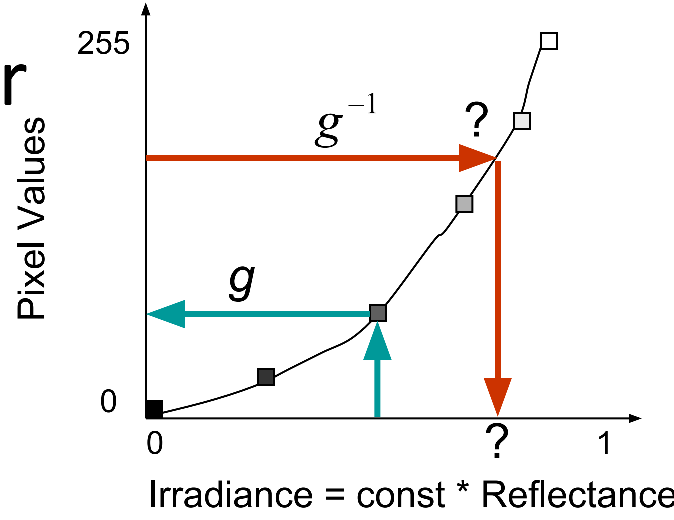
Non-linear Mapping!

Goal: Measured $I(x,y)$ should be "what is out there?"

Need to invert

- Important preprocessing step for many vision and image processing algorithm
- Mental picture: Why the color in different classrooms with different projectors appear different though you as a human can interpret it easily

$$g^{-1} : I \rightarrow E$$



Bottom Line

- What is out there is radiance, and
- If you are doing experiments to measure real physical quantities, make sure to
  - understand the linear (or quasi-linear) relationship
  - inverse mapping
- Some vision algorithms will depend on this step

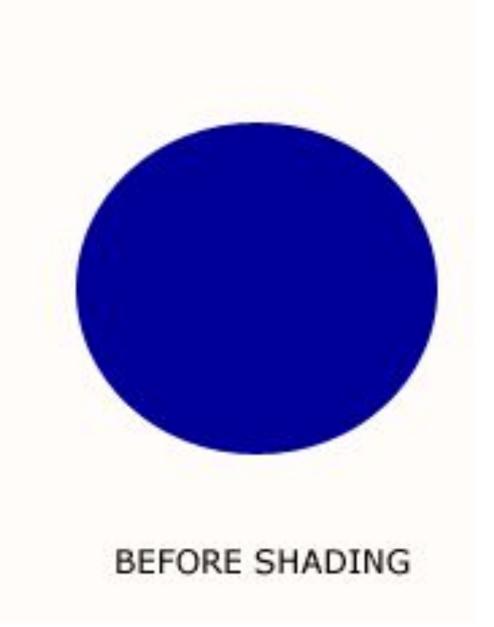
Bottom Line: Pause

- Abstraction: What did we learn? For example
  - mathematics
  - physics
  - computer science

Contents

Problem: Shape from shading

- With the assumption that scene radiance equals image irradiance, given an image *I(x,y)* of a surface, determine the depth at each point on the surface.
  - What is shading, and why would shading reveal the depth?



Recall Example

Depth from Shading: Quantitative

- The depth of an object reveals itself from its surface normal …
- The surface normal reveals itself from the shading because
- Scene radiance is a function of
  - scene illumination
  - material properties
  - local geometry (depth)
  - viewer position
- We have to disentangle these last four things

Contents

- Introduction
- Concept of scene and image irradiance
  - Proving (falsifying) a fundamental assumption
- Problem Definition
- Reflectance function (aka BRDF to be more precise)
- Shape from shading
- Photometric stereo

Problem: Shape from shading

- Given an image $I(x,y)$ of a surface **with a known reflectance model**, determine the depth at each (visible) point on the surface.

Lambertian Model

- The scene radiance is given by the equation:

$$I = L\rho \boldsymbol{l}^T \mathbf{N} = L\rho \cos(\theta),$$

I = scene radiance = image irradiance at the appropriate image point

L = lighting intensity

$\boldsymbol{l}$ = lighting direction (unit vector), assuming distant point light source

$\mathbf{N}$ = unit surface normal at the point under consideration

ρ = surface reflectivity (albedo) at the point under consideration

End of Recap

Contents (original order)

- Introduction
- Concept of scene and image irradiance
  - Proving (falsifying) a fundamental assumption
- Problem Definition
- Reflectance function
- Shape from shading
- Photometric stereo

Contents (Updated Order)

- Introduction
- Concept of scene and image irradiance
  - Proving (falsifying) a fundamental assumption
- Problem Definition
- Reflectance function
- Shape from shading
- Photometric stereo

Photometric Stereo

aka cheating

Photometric stereo

- "Stereo" in computer vision refers to 3D reconstruction from **multiple** images of the **same** object.
- Photometric stereo refers to 3D reconstruction from multiple images of an object under the following conditions:

  ✓ No change in camera position
    - OK to change if Lambertian

  ✓ Change in lighting direction (point light source)

Basic Idea

- Three sources of light represented by three vectors $s_i = [I_{i,x}, I_{i,y}, I_{i,z}]^T$
- For each point $E_i = (ks_i)N(x,y)$ where the normal is unknown
- Write $E = [E_1, E_2, E_3]^T$, $S = [s_{1,}\ s_{2,}\ s_2]^T$, $N=[n_{x,}\ n_{y,}\ n_z]^T$
- Then N can be inverted using a 3x3 matrix inversion
  - albedo is the magnitude of the normal
- More light sources, least squares solution

Idea

- Make sure to see the video …
- More light sources, least squares solution
- There is more to it …

a b

Figure 3.1: Examples of photometric stereo inputs and output. (a) Four raw differently illuminated images. (b) Reconstructions using standard PS.

http://eprints.uwe.ac.uk/16754/
http://www1.uwe.ac.uk/et/mvl/projects/facerecognition.aspx

Hansen, M. (2012) *3D face recognition using photometric stereo.* PhD, University of the West of England.

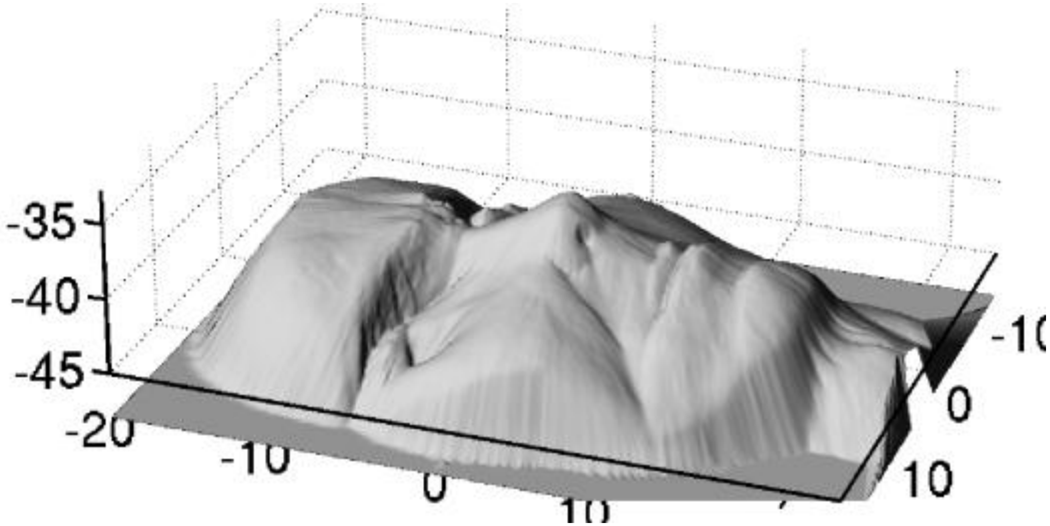
Contents

Depth from Shading: Quantitative

- Scene radiance is a function of
  a. scene illumination (point source, no interreflections)
  b. material properties (Lambertian)
  c. local geometry (depth)
  d. viewer position (far away from object)
- We will use (d) to suitable parametrize (c)

Input Output

- How do we represent depth?



Representing Normal

- The scene is represented by $Z(X,Y)$ in a suitable world coordinate
- However, viewer far away implies we can use the weak perspective camera model
  a. $x = f\, X/Z_0$ and $y = f\, Y/Z_0$ where Z_0 is the average distance from the image plane
  b. Thus the surface can be represented by $Z(x,y)$
- The gradient is represented by $[1, 0, \partial Z/\partial x]^T$ and $[1, 0, \partial Z/\partial y]^T$
- And the normal is?

Representing Normal

The normal is obtained by a cross product

$$(-p,-q,1)=\left(-\frac{\partial Z(x,y)}{\partial x},\quad -\frac{\partial Z(x,y)}{\partial y},\quad 1\right)$$

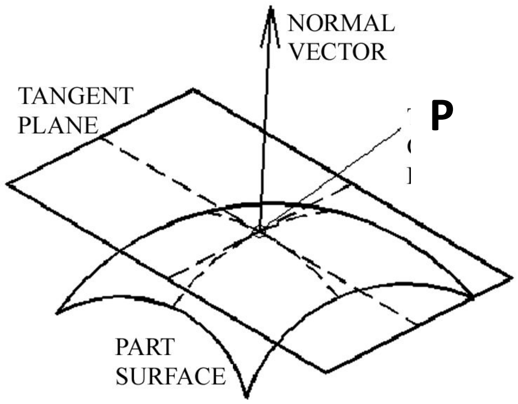
$$=(-p,-q,1)$$

$$=\left(1,0,\frac{\partial Z(x,y)}{\partial x}\right)\times\left(0,1,\frac{\partial Z(x,y)}{\partial y}\right)$$

Normal vector at a point **P** is perpendicular to the tangent plane at the point **P**. The tangent plane at **P** touches the surface only at **P**.

Unit step in X direction induces a change of Z_x in Z.

Unit step in Y direction induces a change of Z_y in Z.



Disentangling

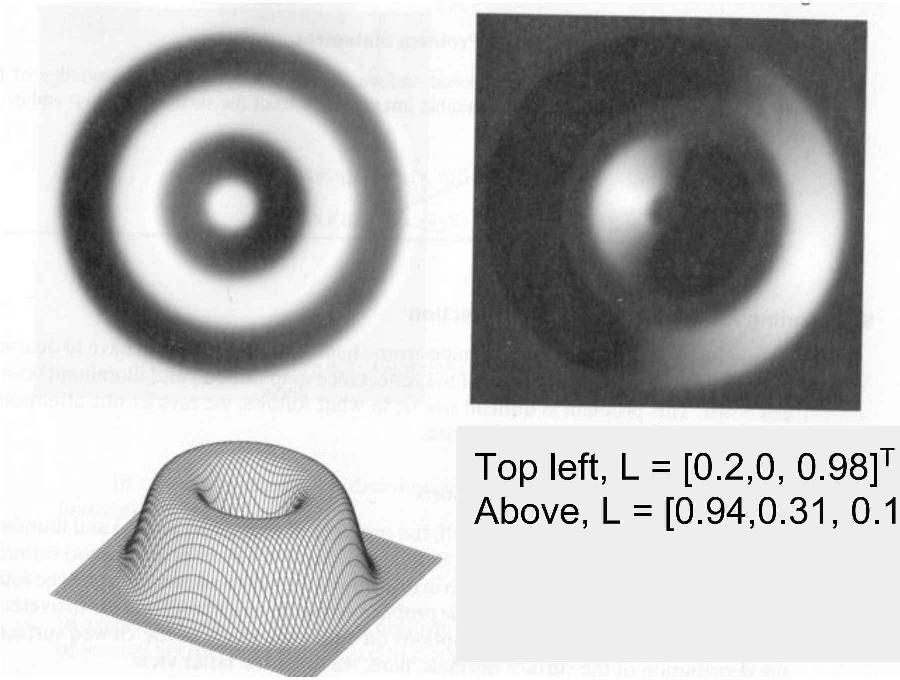
- The scene irradiance is given by the equation:

$$I = L\rho \boldsymbol{l}^T \mathbf{N} = L\rho\cos(\theta) = L\rho \boldsymbol{l}^T \left(\frac{-p}{\sqrt{p^2+q^2+1}}, \quad \frac{-q}{\sqrt{p^2+q^2+1}}, \quad \frac{1}{\sqrt{p^2+q^2+1}} \right)$$

- We know the left hand side
  - we don’t know and want to deduce (p, q)
  - we don’t know the light direction, and the albedo either

Example: Generate Image

- Pick Z(x,y)
- Choose albedo, choose light
- Numerically compute (p, q)
- Plug in equation



Top left, $L = [0.2, 0, 0.98]^T$
Above, $L = [0.94, 0.31, 0.16]^T$

$$I(x,y) = L\rho \frac{p(x,y)l_x + q(x,y)l_y + l_z}{\sqrt{l_x^2 + l_y^2 + l_z^2}\sqrt{p(x,y)^2 + q(x,y)^2 + 1}} = R(p(x,y), q(x,y))$$

Recap: Assumptions

1. **Lambertian reflectance model**
2. The light source is a **point light source**
   – All visible object points receive illumination only from the point light source – there are **no inter-reflections**.
3. The object is somewhat far away relative to its size
   – Weak-perspective model

For now,

– assume the albedo is **known** or has been estimated earlier and is constant across the surface

– light direction is known

Towards Solving Shape From Shading

Shape from Shading

- Problem statement: Given an image *I(x,y)* under all previous assumptions, find *Z(x,y)*.
- At each point, we know intensity value *I(x,y)*, but we need to find *p(x,y)* and *q(x,y)*, given by the equation:

$$I(x,y) = L\rho \frac{p(x,y)l_x + q(x,y)l_y + l_z}{\sqrt{l_x^2 + l_y^2 + l_z^2}\sqrt{p(x,y)^2 + q(x,y)^2 + 1}} = R(p(x,y),q(x,y))$$

- The number of unknowns (*p(x,y)* and *q(x,y)* at each point) is more than the number of knowns (*I(x,y)* at each point)
- For the particular case of Lambertian, with light and albedo know, R(p,q) is known and thus we can compute partial derivatives

Shape from Shading

- We need to impose additional constraints
  - One such constraint is that the underlying surface should be smooth, i.e., the value of *p(x,y)* and the value of *q(x,y)* both change slowly w.r.t. *x* and *y*.
- Hence solve minimization problem:

$$\min \iint_{\Omega} \left[(I(x,y) - R(p(x,y), q(x,y)))^2 + \lambda(p_x^2 + q_x^2 + p_y^2 + q_y^2) \right] dxdy$$

Regularization

- The addition of the smoothness constraint in the under-constrained problem is called as **regularization**! Regularization is a common feature of MANY computer vision problems (and in fact, many problems in machine learning and statistics)
- Emerges from the method of Lagrange multiplier

MOLM

- To minimize f subject to g = 0
  - try minimize f + ƛg
- Consider: Find a point on the surface S: x^2-z^2-$1= 0$ that is closest to the origin
- Requires us to minimize f(x,y,z) = $x^2+y^2+z^2$
- Approach #1: Substitute for z^2

Regularization

$$\min \iint_{\Omega} \Big[\underbrace{(I(x,y) - R(p(x,y), q(x,y)))^2}_{\text{Data fidelity term}} + \underbrace{\lambda}_{\text{Regularization parameter}} \underbrace{(p_x^2 + q_x^2 + p_y^2 + q_y^2)}_{\text{Regularization term (or regularizer)}} \Big] dxdy$$

- The p … a we … een … r and data fidelity term. A larger λ means more weight to the regularizer, and a smaller λ means more weight to the data fidelity term.
- It is usually a user-specified parameters (though there is a large body of literature on automatic choice of λ).

Functionals

$$\min \iint_{\Omega} \left[(I(x,y) - R(p(x,y), q(x,y)))^2 + \lambda (p_x^2 + q_x^2 + p_y^2 + q_y^2) \right] dxdy$$

- In traditional high school calculus, we are given a function and we are looking for a point that minimizes the function
- The term above is called a functional and we are looking for a function (in this case two functions) that minimizes the functional
- Functionals are minimized by the Euler-Lagrange Equations which depends on the form of the functional

Function to optimize	The Euler-Lagrange equations
$\int F(x, u_x) dx$	$F_u - \frac{d}{dx} F_{u_x} = 0$
$\int F(x, u_x, u_{xx}) dx$	$F_u - \frac{d}{dx} F_{u_x} - \frac{d^2}{dx^2} F_{u_{xx}} = 0$
$\int F(x, u_x, v_x) dx$	$F_u - \frac{d}{dx} F_{u_x} = 0$ $F_v - \frac{d}{dx} F_{v_x} = 0$
$\int \int F(x, y, u_x, u_y) dxdy$	$F_u - \frac{d}{dx} F_{u_x} - \frac{d}{dy} F_{u_y} = 0$

Vision and E-L

Problem	*Regularization principle*
Contours	$\int E_{snake}(\mathbf{v}(s))ds$
Area based Optical flow	$\int[({u_x}^2+{u_y}^2+{v_x}^2+{v_y}^2)+\lambda(E_x u+E_y v+i_t)^2)]dxdy$
Edge detection	$\int[(Sf-i)^2+\lambda\,(f_{xx})^2]dx$
Contour based Optical flow	$\int[(V\cdot N-V^N)^2+\lambda(\frac{\delta V}{\delta x})^2]$
Surface reconstruction	$\int[(S\cdot f-d^2+\lambda(f_{xx}+2{f_{xy}}^2+{f_{yy}}^2)]dxdy$
Spatiotemporal approximation	$\int[(S\cdot f-i)^2+\lambda(\nabla f\cdot V+ft)^2]dxdydt$
Colour	$\|I^y-Ax\|^2+\lambda\|Pz\|^2$
Shape from shading	$\int[(E-R(f,g))^2+\lambda({f_x}^2+{f_y}^2+{g_x}^2+{g_y}^2)]dxdy$
Stereo	$\int\{[\nabla^2 G*(L(x,y)-R(x+d(x,y),y))]^2+\lambda(\nabla d)^2\}dxdy$

Solution

- Applying the E-L equation we get

Laplacian of p (or q)

$$-(I(i,j)-R(p(i,j),q(i,j))\frac{\partial R}{\partial p(i,j)}=\lambda\boxed{(p_{xx}(i,j)+p_{yy}(i,j))}$$

$$-(I(i,j)-R(p(i,j),q(i,j)))\frac{\partial R}{\partial q(i,j)}=\lambda\boxed{(q_{xx}(i,j)+q_{yy}(i,j))}$$

discretization

$$(p(i+1,j)+p(i-1,j)+p(i,j+1)+p(i,j-1)-4p(i,j))$$
$$(q(i+1,j)+q(i-1,j)+q(i,j+1)+q(i,j-1)-4q(i,j))$$

Solution

- Discretizing we get

$$-(I(i,j)-R(p(i,j),q(i,j))\frac{\partial R}{\partial p(i,j)} = \lambda(p(i+1,j+1)+p(i+1,j-1)+p(i-1,j)+p(i-1,j-1)-4p(i,j))$$

$$-(I(i,j)-R(p(i,j),q(i,j))\frac{\partial R}{\partial q(i,j)} = \lambda(q(i+1,j+1)+q(i+1,j-1)+q(i-1,j)+q(i-1,j-1)-4q(i,j))$$

- Re-arranging the terms, we get:

$$p(i,j) = \frac{1}{4}(p(i+1,j+1)+p(i+1,j-1)+p(i-1,j)+p(i-1,j-1)) + \frac{(I(i,j)-R(p(i,j),q(i,j))\frac{\partial R}{\partial p(i,j)}}{\lambda}$$

$$q(i,j) = \frac{1}{4}(q(i+1,j+1)+q(i+1,j-1)+q(i-1,j)+q(i-1,j-1)) + \frac{(I(i,j)-R(p(i,j),q(i,j))\frac{\partial R}{\partial q(i,j)}}{\lambda}$$

Algorithm: SfS

- For a fixed number of iterations *T* (or until a convergence criterion is met), with *t* being the iteration number:

$$\forall(i,j)\in\Omega, p(i,j)\leftarrow 0, q(i,j)\leftarrow 0$$

$$for(t=1:T)$$

$$\{$$

$$p^{(t+1)}(i,j)=\frac{1}{4}(p^{(t)}(i+1,j+1)+p^{(t)}(i+1,j-1)+p^{(t)}(i-1,j)+p^{(t)}(i-1,j-1))$$

$$+\frac{1}{\lambda}(I(i,j)-R(p^{(t)}(i,j),q^{(t)}(i,j))\boxed{\frac{\partial R}{\partial p^{(t)}(i,j)}}$$

Evaluate the expression for the derivative using older values, i.e. $p^{(t)}(i,j)$ and $q^{(t)}(i,j)$

$$q^{(t+1)}(i,j)=\frac{1}{4}(q^{(t)}(i+1,j+1)+q^{(t)}(i+1,j-1)+q^{(t)}(i-1,j)+q^{(t)}(i-1,j-1))$$

$$+\frac{1}{\lambda}(I(i,j)-R(p^{(t)}(i,j),q^{(t)}(i,j))\boxed{\frac{\partial R}{\partial q^{(t)}(i,j)}}$$

Evaluate the expression for the derivative using older values, i.e. $p^{(t)}(i,j)$ and $q^{(t)}(i,j)$

$$\}$$

Depth from needle map

- Knowledge of $p(x,y)$ and $q(x,y)$ gives us surface normals at each point. The set of all surface normals at visible points on the object is called the **needle map**.

- How to find the depth $Z(x,y)$ at each point?

1-D version

- Let us suppose that we had a 1-D function, i.e., we have *Z(x)* instead of *Z(x,y)*.
  - Given *Z(x)* at each point, it is trivial to find the (discrete) derivative at each point, i.e. *Z'(x) = Z(x+1)-Z(x)*.
- Given *Z'(x)*, we can estimate *Z(x)* by integration – **up to an unknown constant of integration**.
- The unknown constant should not bother us

6	5	8	4	0

23+*a*	17+*a*	12+*a*	4+*a*	0+*a*

2D Version

Values of p(x,y) (i.e.
x-derivatives of depth)

6	7	8	0
7	1	1	0
8	3	2	0
3	4	6	0

Values of q(x,y) (i.e.
y-derivatives of depth)

4	7	2	5
3	3	9	4
6	8	6	3
0	0	0	0

- Integrating the p values across each row produces depth (i.e. Z) values.
- Integrating the q values across each column also produces depth values.
- The two estimates of depth values may not be consistent!

Remarks

- Good: *R(p,q)* is any surface reflectance map
  - May be something different from Lambertian. The general theory for SfS still holds
- Bad:
  - Can't predict λ, T, rate of convergence
- Very Bad:
  - p and q are obtained independently
  - no guarantee that $Z_{xy} = Z_{yx}$

Integrability Constraint in SfS

- Fundamental issue with this approach. It treats *p* and *q* as independent quantities when in reality, we know that:

$$p = \frac{\partial z}{\partial x}, q = \frac{\partial z}{\partial y}$$

- Furthermore, if the partial derivatives are continuous, it turns out that:

$$\frac{\partial p}{\partial y} = \frac{\partial q}{\partial x} (= \frac{\partial^2 z}{\partial x \partial y} = \frac{\partial^2 z}{\partial y \partial x}) \Rightarrow \boxed{\text{Integrability Constraint}}$$

Least squares to the rescue

- We can instead find a depth map **Z(x,y)** which satisfies the following:

$$\forall(x,y), \frac{\partial Z(x,y)}{\partial x} \approx p(x,y) \text{ and } \frac{\partial Z(x,y)}{\partial y} \approx q(x,y)$$

Known *p* and *q*, unknown *Z*

- Hence, we find a *Z(x,y)* that minimizes the following:

$$J(Z) = \iint_{\Omega} \left[\left(\frac{\partial Z(x,y)}{\partial x} - p(x,y) \right)^2 + \left(\frac{\partial Z(x,y)}{\partial y} - q(x,y) \right)^2 \right] dxdy$$

Given data: p(x,y) and q(x,y)
Unknown data: Z(x,y)

Poisson equation

- Using E-L formulation, we get an instance of the **Poisson equation**

$$Z_{xx}(x,y) + Z_{yy}(x,y) = s(x,y)$$

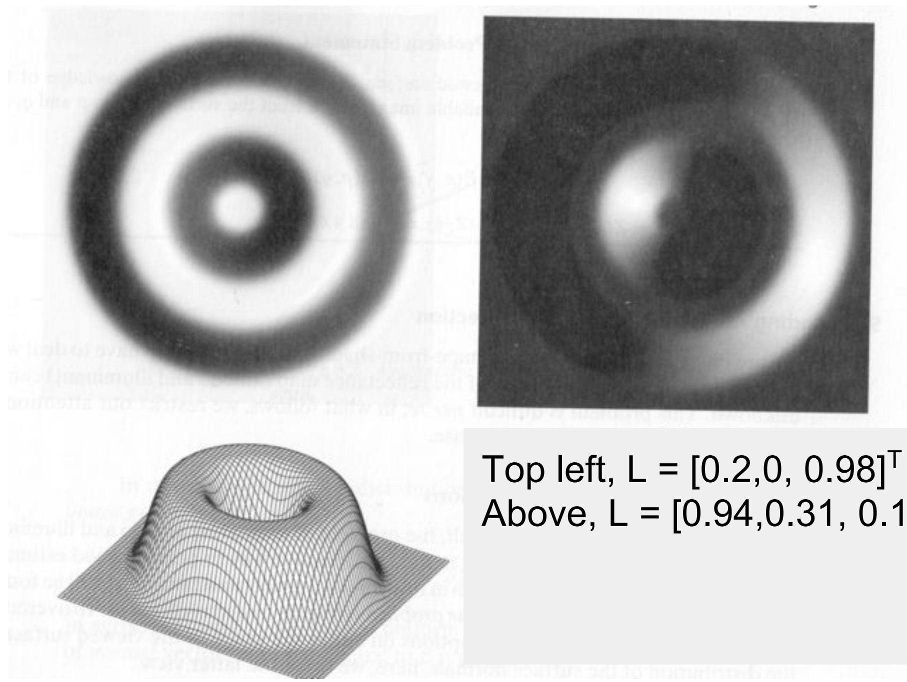
- This equation can be solved using the Discrete Fourier Transform (out of scope/time, but see next slide for idea)

Remarks

- Good: *R(p,q)* is any surface reflectance map
  - May be something different from Lambertian. The general theory for SfS still holds
- Bad:
  - Can't predict λ, T, rate of convergence
- Very Bad:
  - p and q are obtained independently, no guarantee that $Z_{xy} = Z_{yx}$
- Fix: Project p and q onto a set which contains only integrable pair (e.g. FFT of (p,q), and then differentiate)

Recall Example

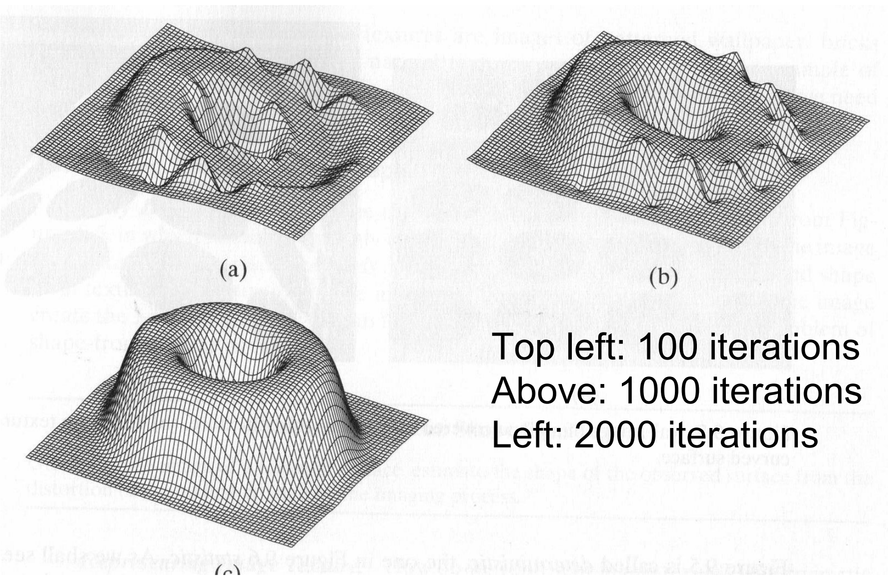
- Pick Z(x,y)
- Choose albedo, choose light
- Numerically compute (p, q)
- Plug in equation



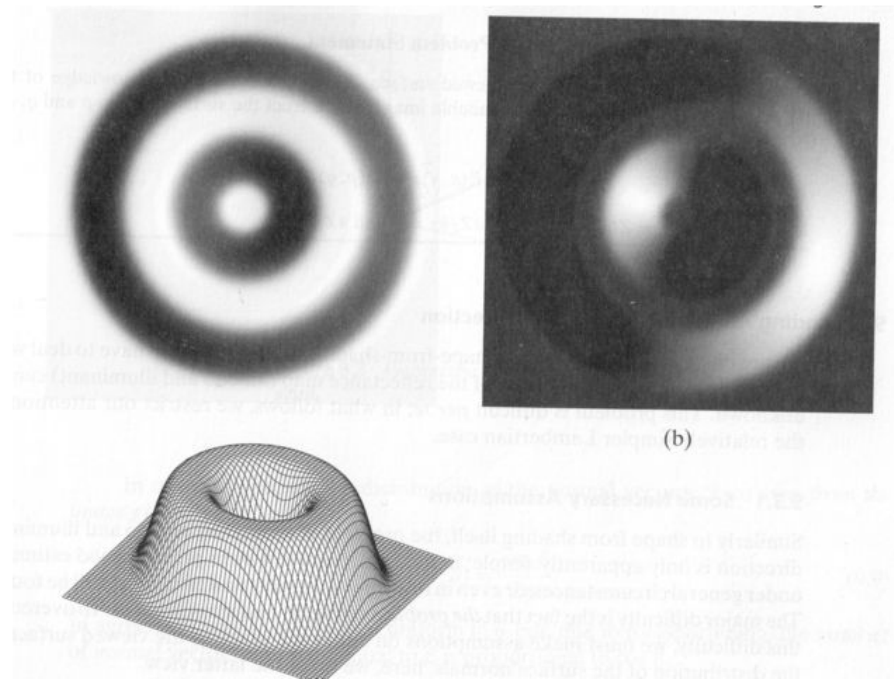
Top left, $L = [0.2, 0, 0.98]^T$
Above, $L = [0.94, 0.31, 0.16]^T$

$$I(x,y) = L\rho \frac{p(x,y)l_x + q(x,y)l_y + l_z}{\sqrt{l_x^2 + l_y^2 + l_z^2}\sqrt{p(x,y)^2 + q(x,y)^2 + 1}} = R(p(x,y), q(x,y))$$

Recall Example



Top left: 100 iterations
Above: 1000 iterations
Left: 2000 iterations



$$I(x,y) = L\rho \frac{p(x,y)l_x + q(x,y)l_y + l_z}{\sqrt{l_x^2 + l_y^2 + l_z^2}\sqrt{p(x,y)^2 + q(x,y)^2 + 1}} = R(p(x,y), q(x,y))$$

Summary: Recall Agenda

- Want: Reconstruct three dimensional structure from a photo (or 3 photos)
- Why
  - Relighting, generate new view, generate expressions, **generate feature vectors for ML**
  - Fundamental problem in vision
- How this is approached
  - We will take a “first principles approach”
  - Some (possibly) new mathematical concepts