Photometric Stereo

Computer Vision I

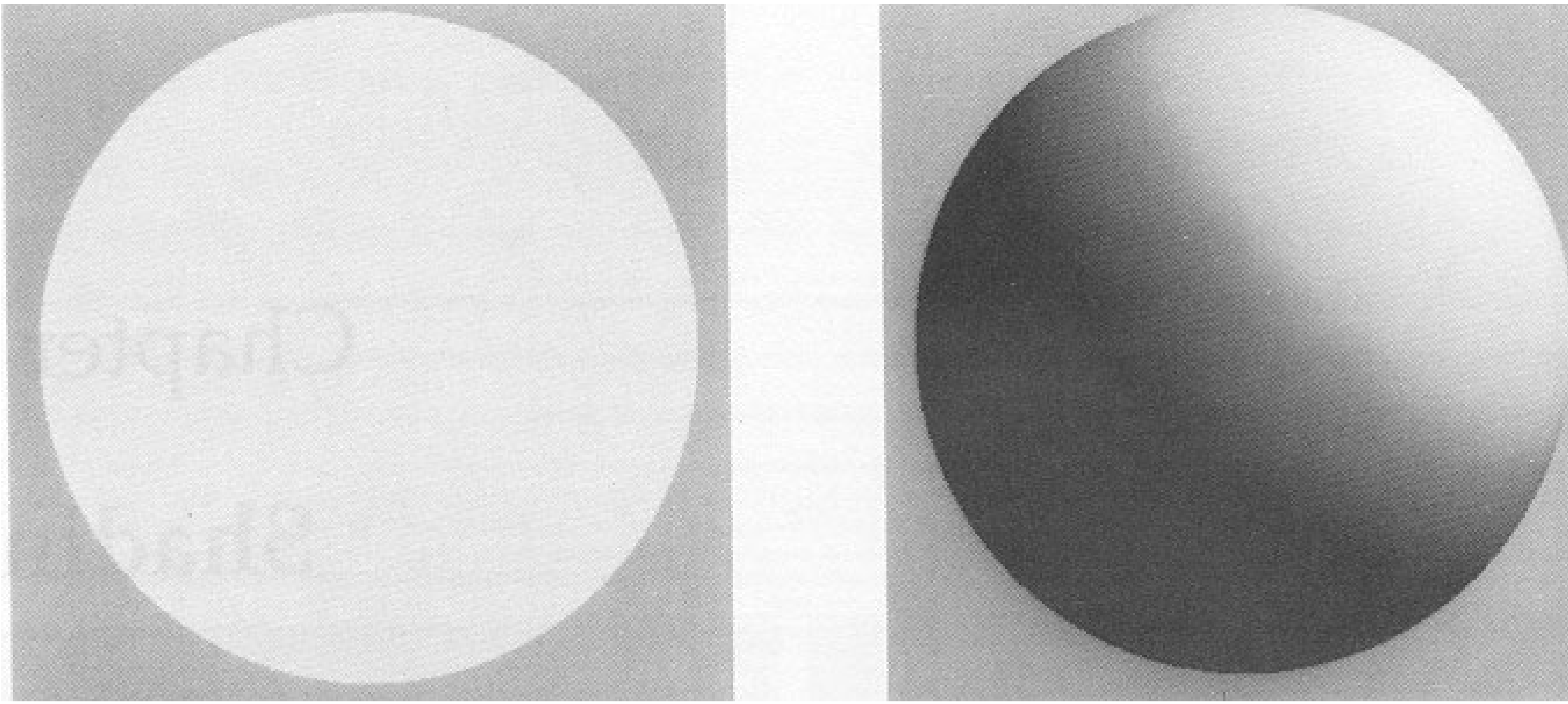
CSE 252A

Lecture 4

Announcements

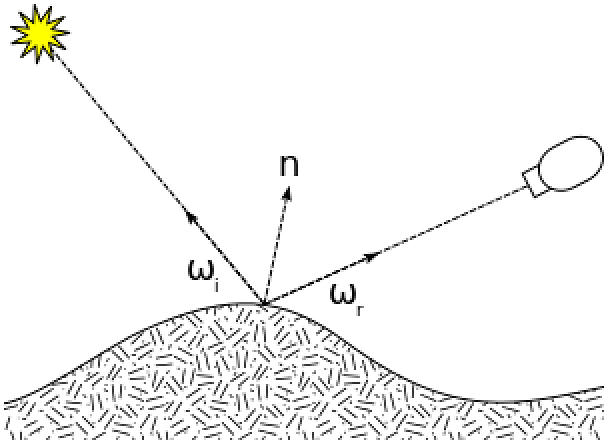
- Assignment 0 is due today, 11:59 PM
- Assignment 1 will be released today
  - Due Oct 25, 11:59 PM

Shading reveals 3D surface geometry



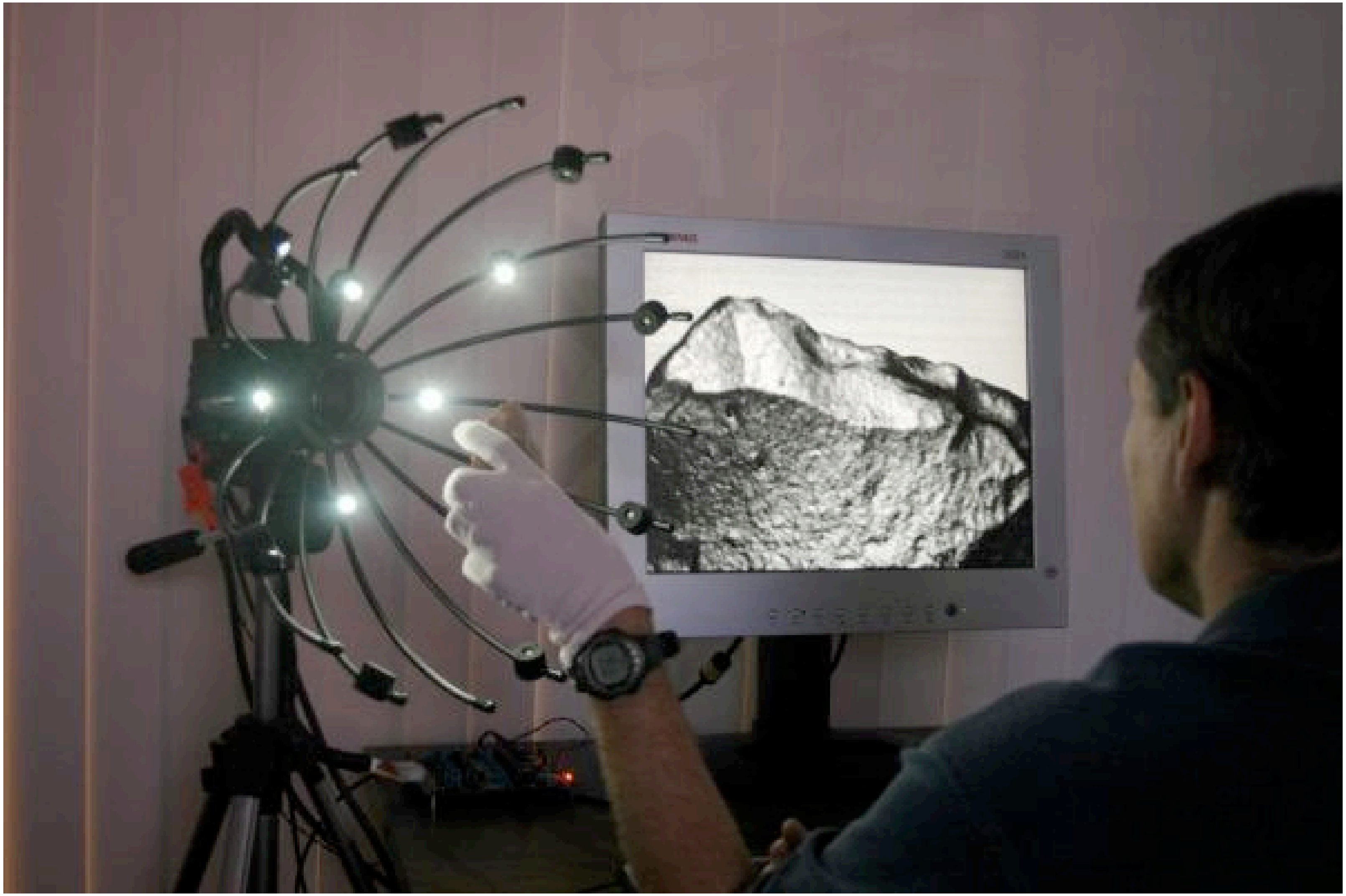
Two shape-from-X methods that use shading

- Shape-from-shading: Use just one image to recover shape. Requires knowledge of light source direction and BRDF everywhere. Too restrictive (good candidate for deep learning)
- Photometric stereo: Single viewpoint, multiple images under different lighting.

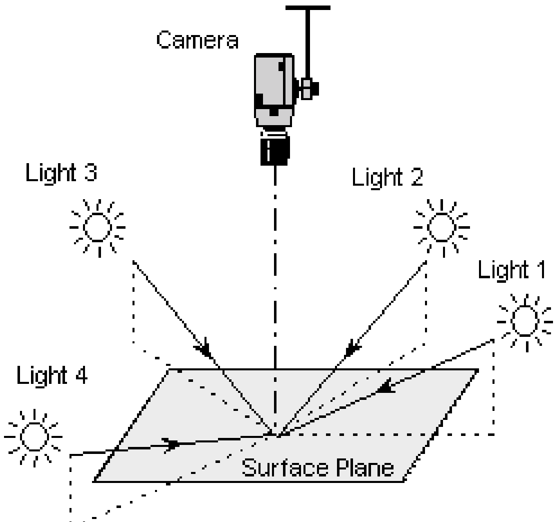


BRDF
(four dimensional function)

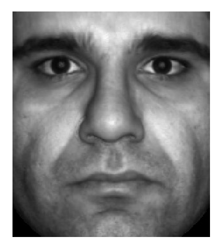
Photometric Stereo Rigs: One viewpoint, changing lighting



An example of photometric stereo



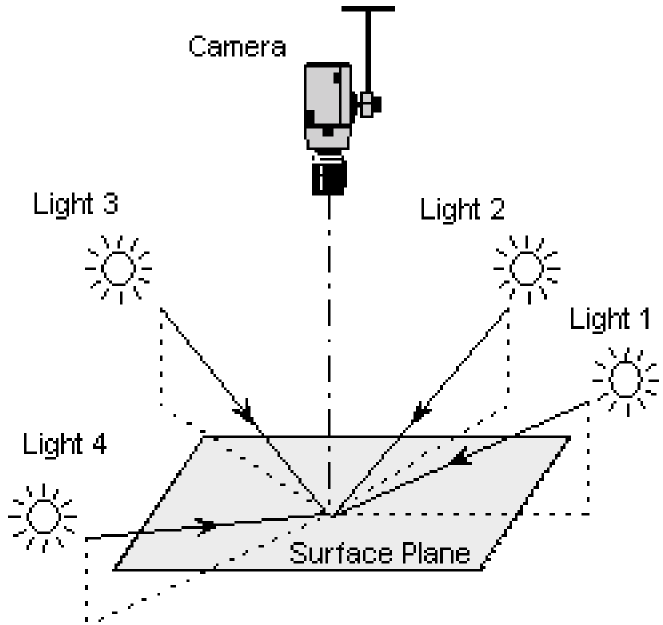
surface
(albedo texture map)



+ surface normals

albedo

Photometric stereo



- Single viewpoint, multiple images under different lighting
  1. General BRDF, known lighting
  2. Lambertian BRDF, known lighting
  3. Lambertian BRDF, unknown lighting

1. Photometric Stereo: General BRDF and Reflectance Map

BRDF

- Bi-directional Reflectance Distribution Function

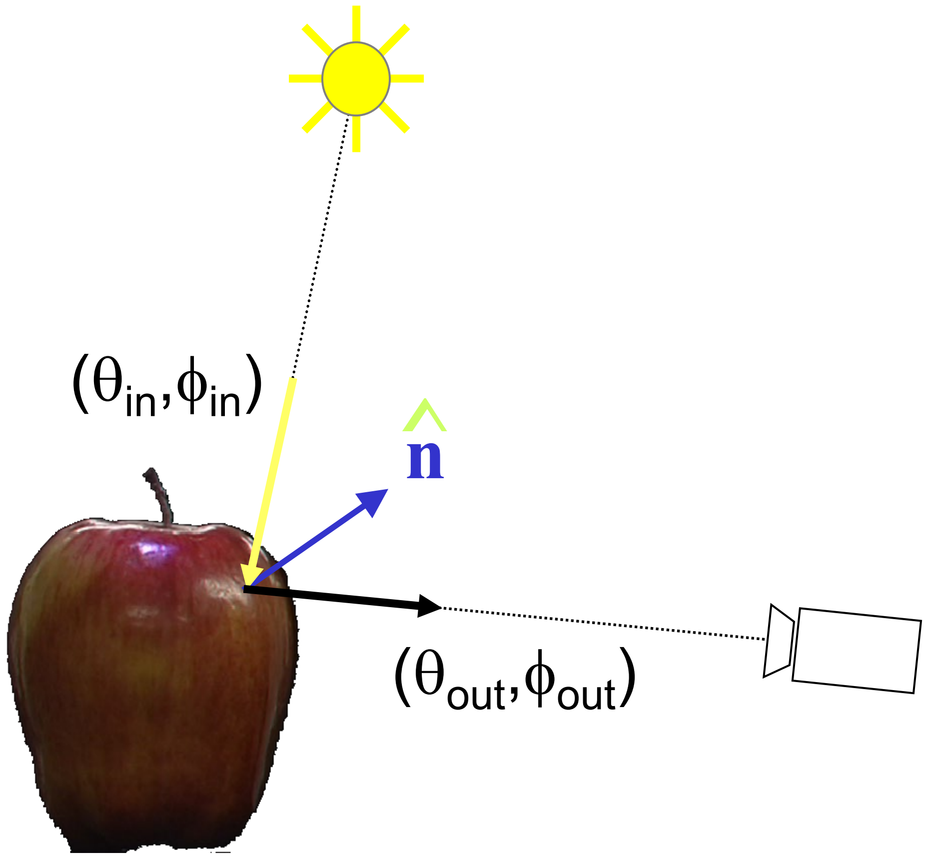
  $$\rho(\theta_{in}, \phi_{in} ; \theta_{out}, \phi_{out})$$

- Function of
  - Incoming light direction:

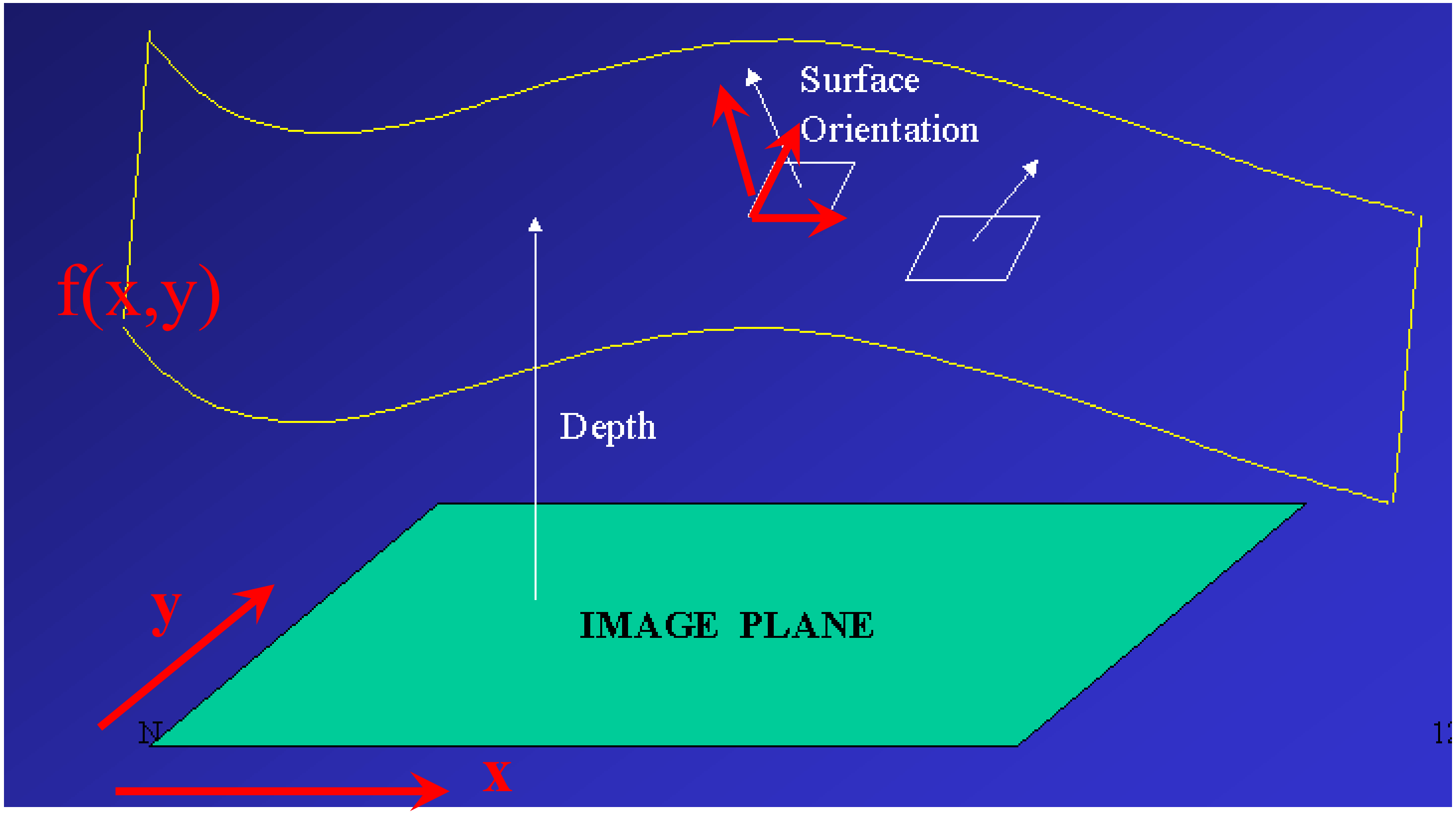
    $$\theta_{in} , \phi_{in}$$
  - Outgoing light direction:

    $$\theta_{out} , \phi_{out}$$

- Ratio of emitted radiance to incident irradiance



Coordinate system



Surface: $\mathbf{s}(x,y) = (x, y, f(x,y))$

Tangent vectors:
$$\frac{\partial s(x,y)}{\partial x} = \left(1, 0, \frac{\partial f}{\partial x}\right)$$

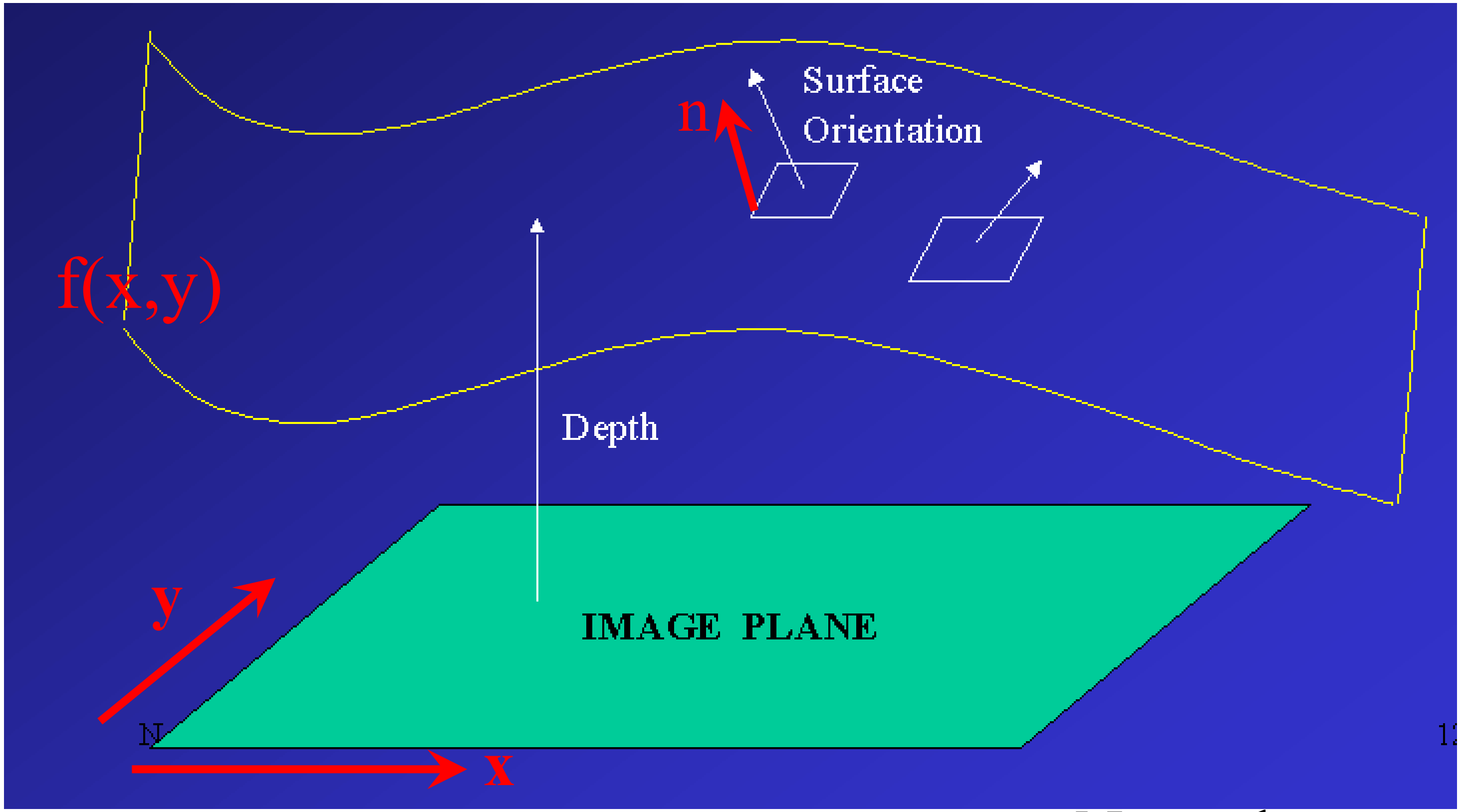
$$\frac{\partial s(x,y)}{\partial y} = \left(0, 1, \frac{\partial f}{\partial y}\right)$$

Normal vector

$$\mathbf{n} = \frac{\partial s}{\partial x} \times \frac{\partial s}{\partial y}$$

$$= \left(-\frac{\partial f}{\partial x}, -\frac{\partial f}{\partial y}, 1\right)$$

Gradient Space (p,q)



Normal vector

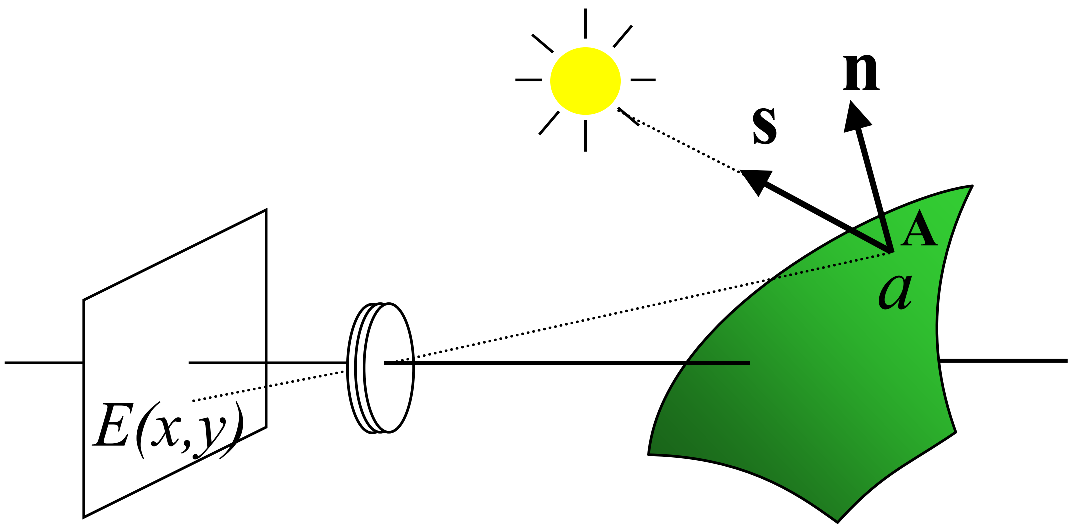
$$\mathbf{n} = \frac{\partial s}{\partial x} \times \frac{\partial s}{\partial y} = \left(-\frac{\partial f}{\partial x}, -\frac{\partial f}{\partial y}, 1 \right)^T$$

$$\hat{\mathbf{n}} = \frac{1}{\sqrt{p^2 + q^2 + 1}} \left(-p, -q, 1 \right)^T$$

Gradient Space : (p,q)

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

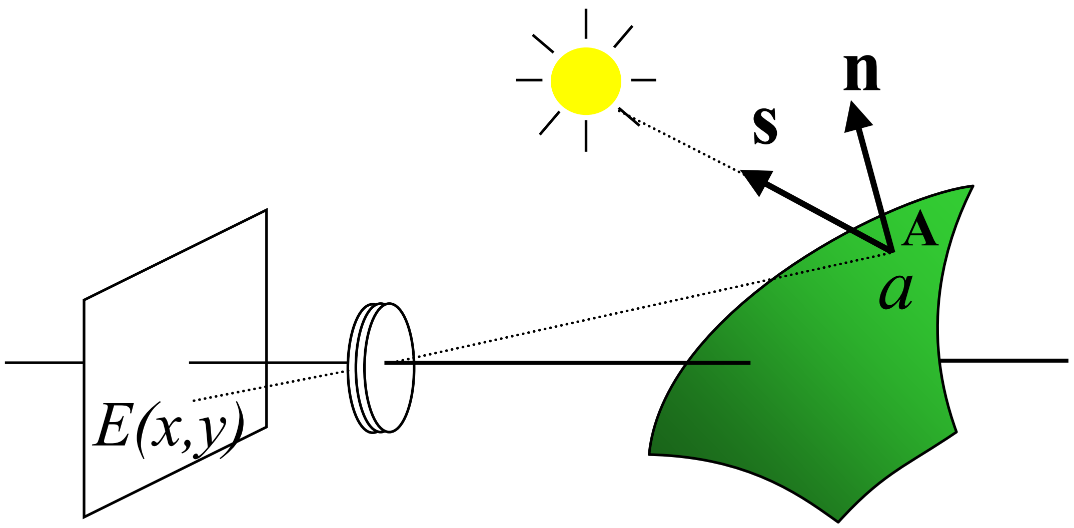
Image Formation



For a given point A on the surface *a*, the image irradiance E(x,y) is a function of

1. The BRDF at A
2. The surface normal at A
3. The direction of the light source

Reflectance Map

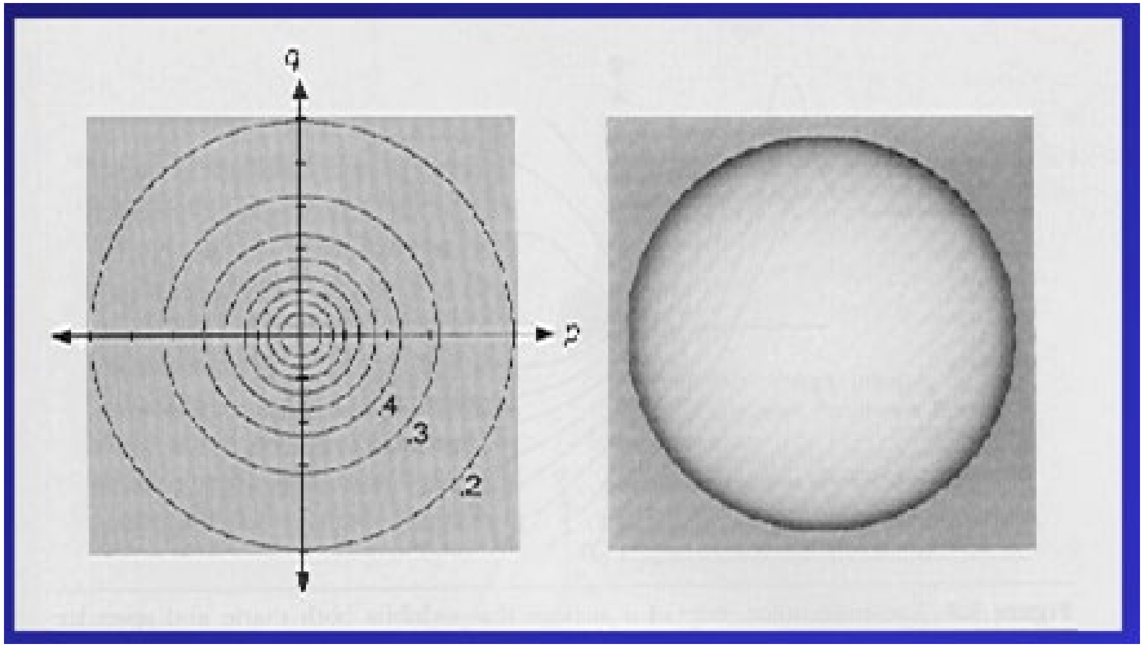


Let the BRDF be the same at all points on the surface, and let the light direction **s** be a constant

1. Then image irradiance is a function of only the direction of the surface normal
2. In gradient space, we have E(p,q)

Example Reflectance Map: Lambertian surface

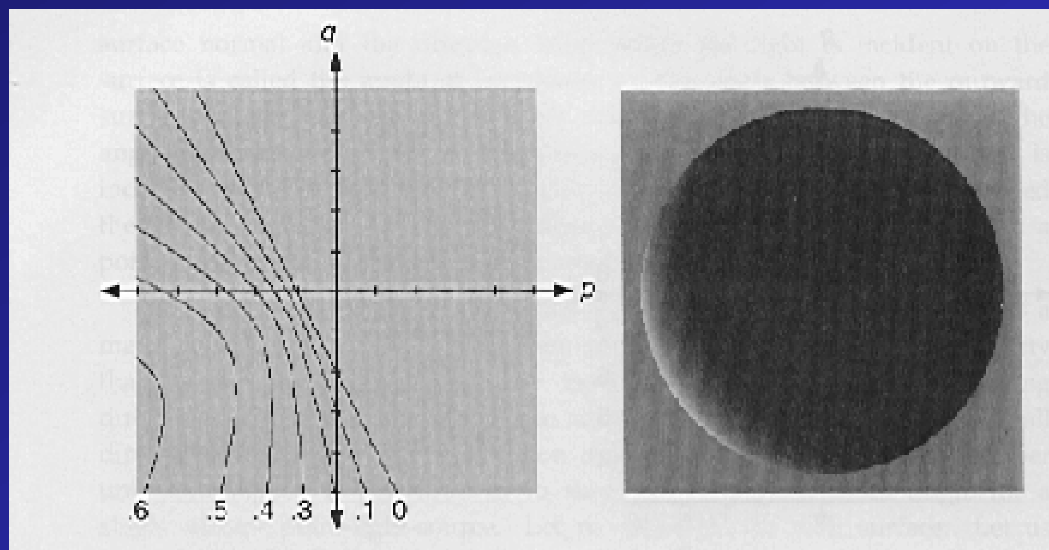
E(p,q)



For lighting from front

LAMBERTIAN REFLECTANCE MAP

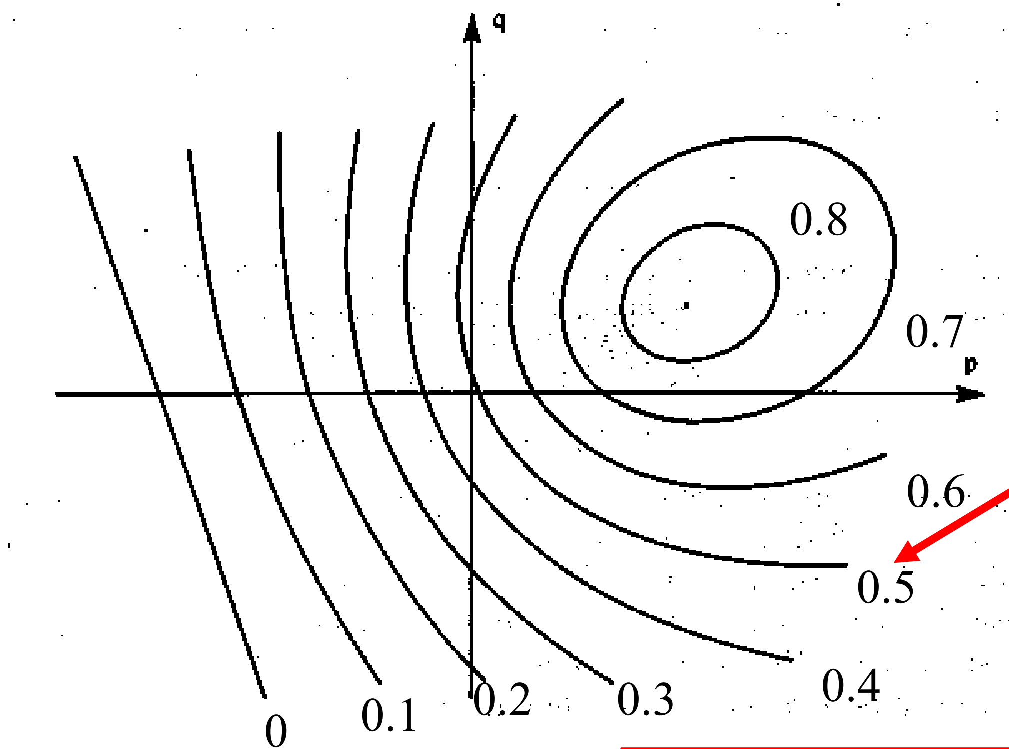
$$E = L\rho \frac{1 + pp_s + qq_s}{\sqrt{1+p^2+q^2}\sqrt{1+p_s{}^2+q_s{}^2}}$$

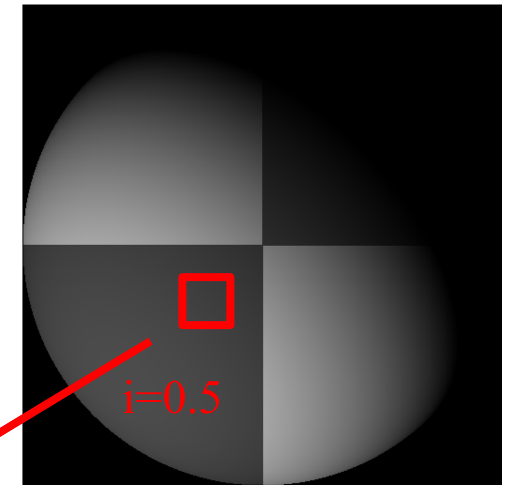


$p_s = -2 \quad q_s = -1$

Light Source Direction, expressed in gradient space.

Reflectance Map of Lambertian Surface





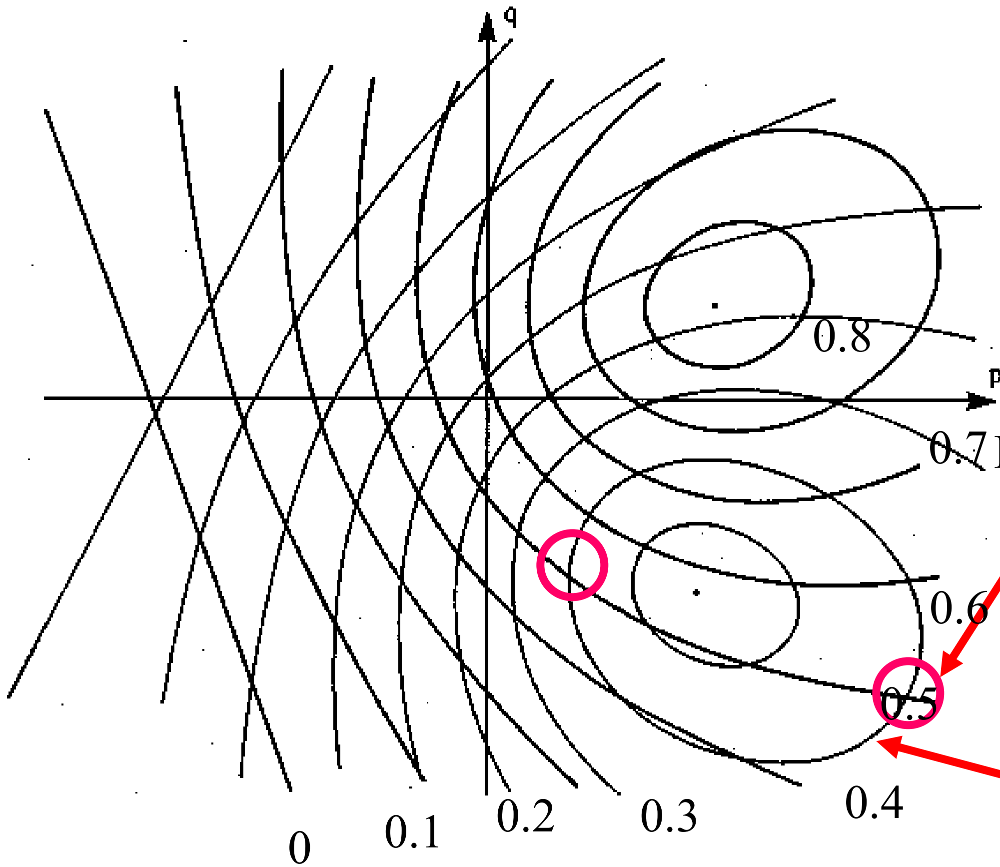
E.g., Normal lies on this curve

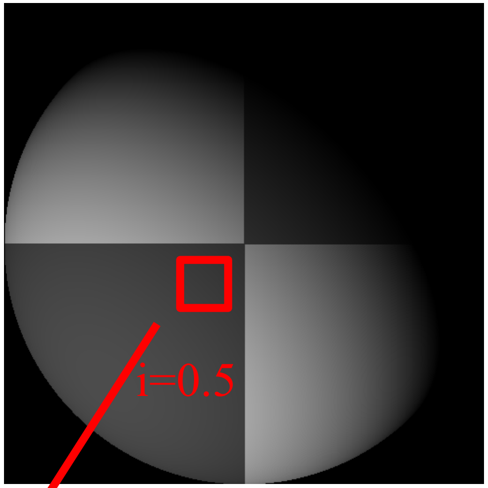
Curves are not really for this image

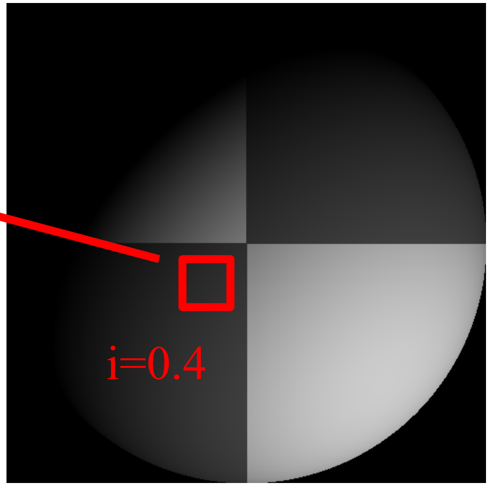
What does the intensity (irradiance) of one pixel in one image tell us?

It constrains the surface normal projecting to that point to a curve

Two Light Sources, Two reflectance maps







Curves are not really for these images

A third image would disambiguate match

Three Source Photometric stereo: Step1

Offline:

Using source directions & BRDF, construct reflectance map for each light source direction. $R_1(p,q)$, $R_2(p,q)$, $R_3(p,q)$

Online:

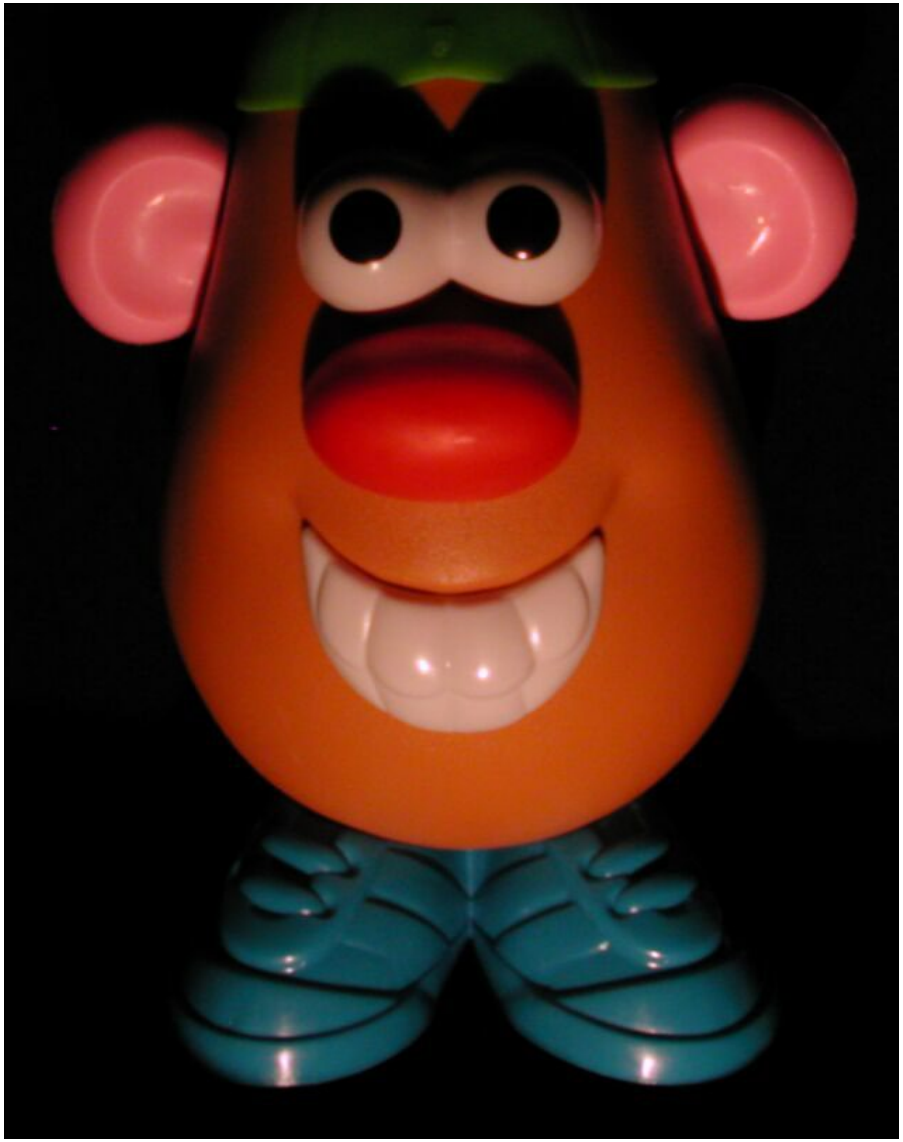
1. Acquire three images with known light source directions. $E_1(x,y)$, $E_2(x,y)$, $E_3(x,y)$
2. For each pixel location (x,y), find (p,q) as the intersection of the three curves

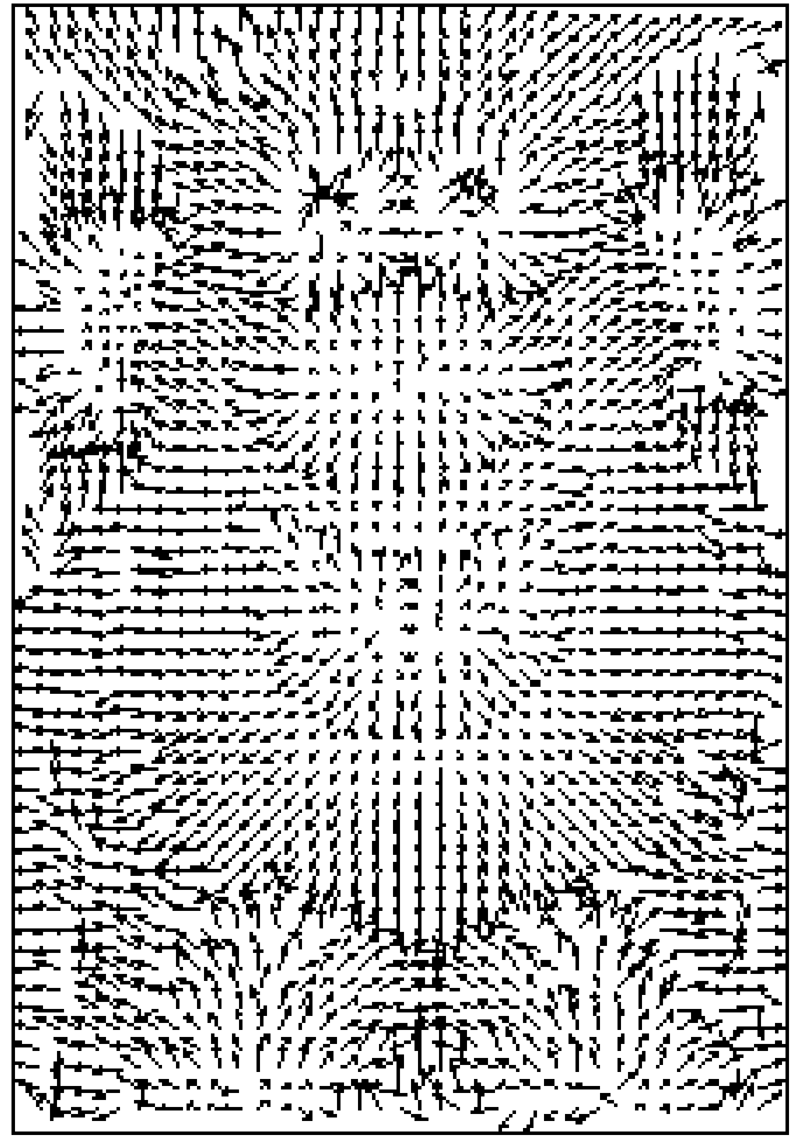
   $R_1(p,q)=E_1(x,y)$

   $R_2(p,q)=E_2(x,y)$

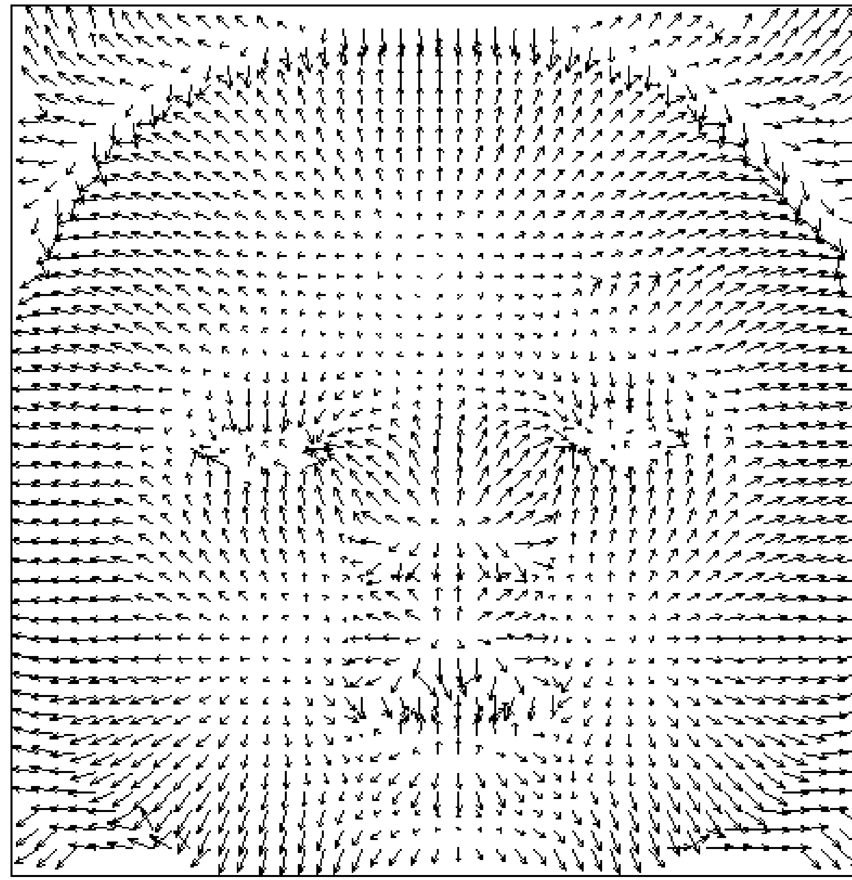
   $R_3(p,q)=E_3(x,y)$
3. This is the surface normal at pixel (x,y). Over image, the normal field is estimated

Normal Field

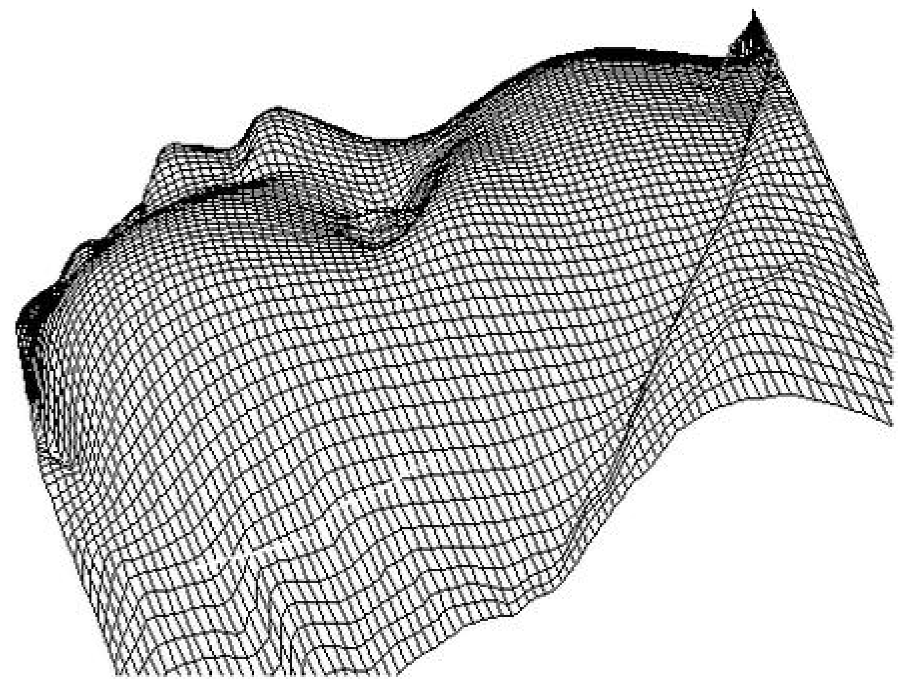




Plastic Baby Doll: Normal Field



Next step:
Go from normal field to surface

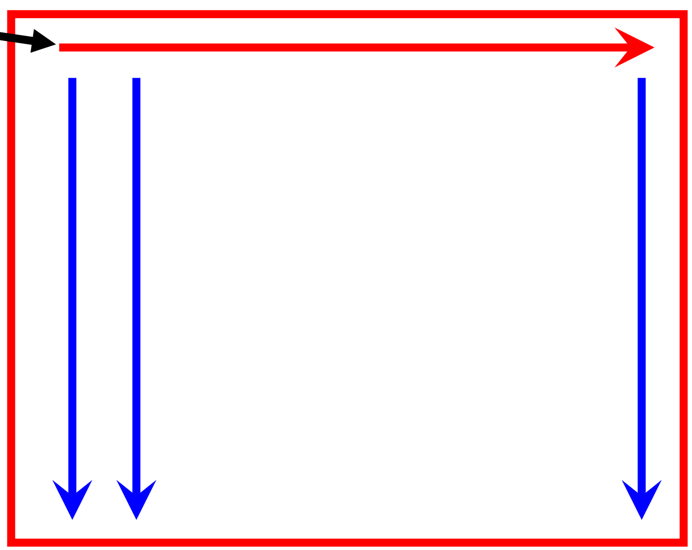


Recovering the surface f(x,y)

Many methods: Simplest approach

1. From estimate **n** $=(n_x,n_y,n_z)$, $p=-n_x/n_z$, $q=-n_y/n_z$
2. Integrate p=df/dx along row (x,0) to get f(x,0)
3. Then integrate q=df/dy along each column starting with value of the first row

Start here, setting f(0,0) = 0

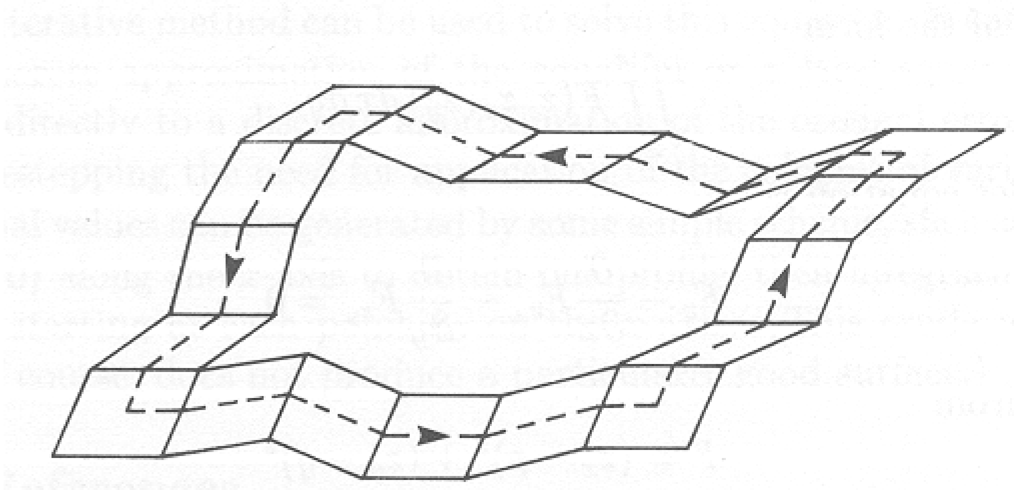


f(x,0)

Use a convex mask to only include object

When done, some values may be negative. Subtract all values from the minimum value.

What might go wrong?



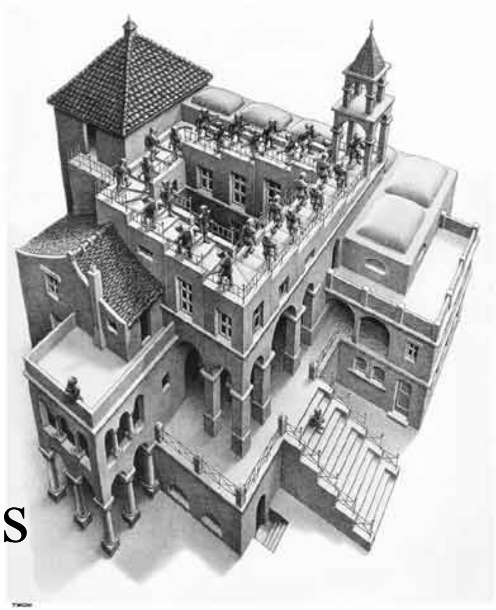
- Height z(x,y) is obtained by integration along a curve from (x_0, y_0).

$$z(x, y) = z(x_0, y_0) + \int_{(x_0, y_0)}^{(x, y)} (p\,dx + q\,dy)$$

- If one integrates the derivative field along any closed curve, one expects to get back to the starting value.

- Might not happen because of noisy estimates of (p,q)

What might go wrong?



Integrability. If f(x,y) is the height function, we expect that

$$\frac{\partial}{\partial y}\frac{\partial f}{\partial x} = \frac{\partial}{\partial x}\frac{\partial f}{\partial y}$$

In terms of estimated gradient space (p,q), this means:

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

But since p and q were estimated indpendently at each point as intersection of curves on three reflectance maps, equality is not going to exactly hold

Horn's Method

["Robot Vision", B.K.P. Horn, 1986]

- Formulate estimation of surface height z(x,y) from gradient field by minimizing cost functional:

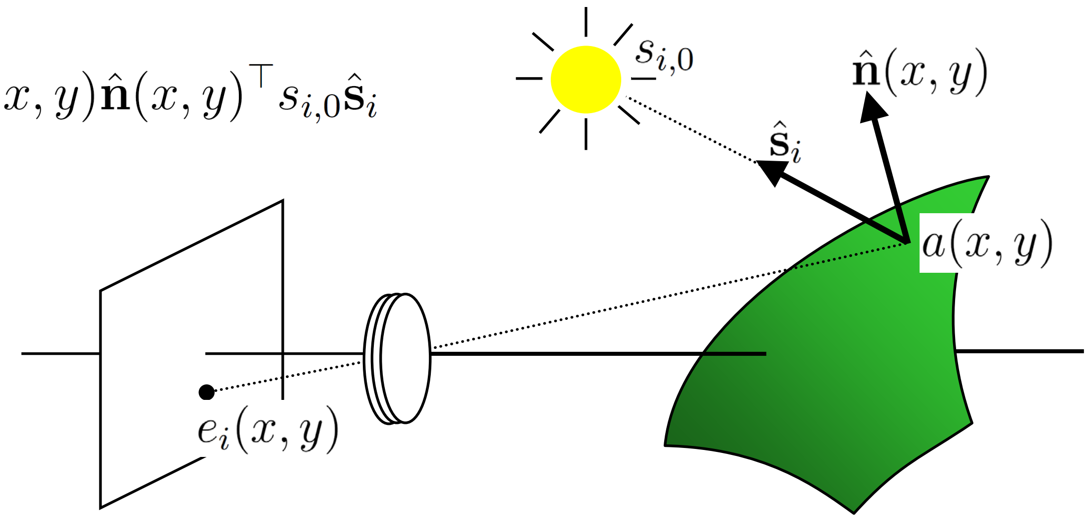
$$\iint_{\text{Image}} (z_x - p)^2 + (z_y - q)^2 \, dxdy$$

  where (p,q) are estimated components of the gradient while z_x and z_y are partial derivatives of best fit surface
- Solved using calculus of variations – iterative updating
- z(x,y) can be discrete or represented in terms of basis functions
- Integrability is naturally satisfied

2. Photometeric Stereo: Lambertian Surface, Known Lighting

Photometric stereo, Lambertian surface

$$e_i(x,y) = a(x,y)\hat{\mathbf{n}}(x,y)^\top s_{i,0}\hat{\mathbf{s}}_i$$



$s_{i,0}$ is the intensity of the i-th light source
$\hat{\mathbf{s}}_i$ is the i-th light source direction at the surface point
$\hat{\mathbf{n}}(x,y)$ is the normal at the surface point projected to the image coordinates (x,y)
$a(x,y)$ is the albedo of the surface point projected to the image coordinates (x,y)
$e_i(x,y)$ is the intensity of the light reflected from the surface point projected to the i-th image coordinates (x,y)

Photometric stereo, Lambertian surface

For each pixel

$e_i(x,y) = a(x,y)\hat{\mathbf{n}}(x,y)^\top s_{i,0}\hat{\mathbf{s}}_i$, solve for $a(x,y)$ and $\hat{\mathbf{n}}(x,y)$

$e_i(x,y) = \mathbf{b}(x,y)^\top \mathbf{s}_i$, where $\mathbf{b}(x,y) = a(x,y)\hat{\mathbf{n}}(x,y)$ and $\mathbf{s}_i = s_{i,0}\hat{\mathbf{s}}_i$

$e_i(x,y) = \mathbf{s}_i^\top \mathbf{b}(x,y)$

$$\begin{bmatrix} e_1(x,y) \\ e_2(x,y) \\ \vdots \\ e_n(x,y) \end{bmatrix} = \begin{bmatrix} \mathbf{s}_1^\top \\ \mathbf{s}_2^\top \\ \vdots \\ \mathbf{s}_n^\top \end{bmatrix} \mathbf{b}(x,y)$$

$n \geq 3$ light sources and associated images

$$\mathbf{e}(x,y) = \mathtt{S}\mathbf{b}(x,y), \text{ solve for } \mathbf{b}(x,y)$$

$$\mathtt{S}^\top \mathbf{e}(x,y) = \mathtt{S}^\top \mathtt{S}\mathbf{b}(x,y)$$

$$(\mathtt{S}^\top \mathtt{S})^{-1}\mathtt{S}^\top \mathbf{e}(x,y) = \mathbf{b}(x,y)$$

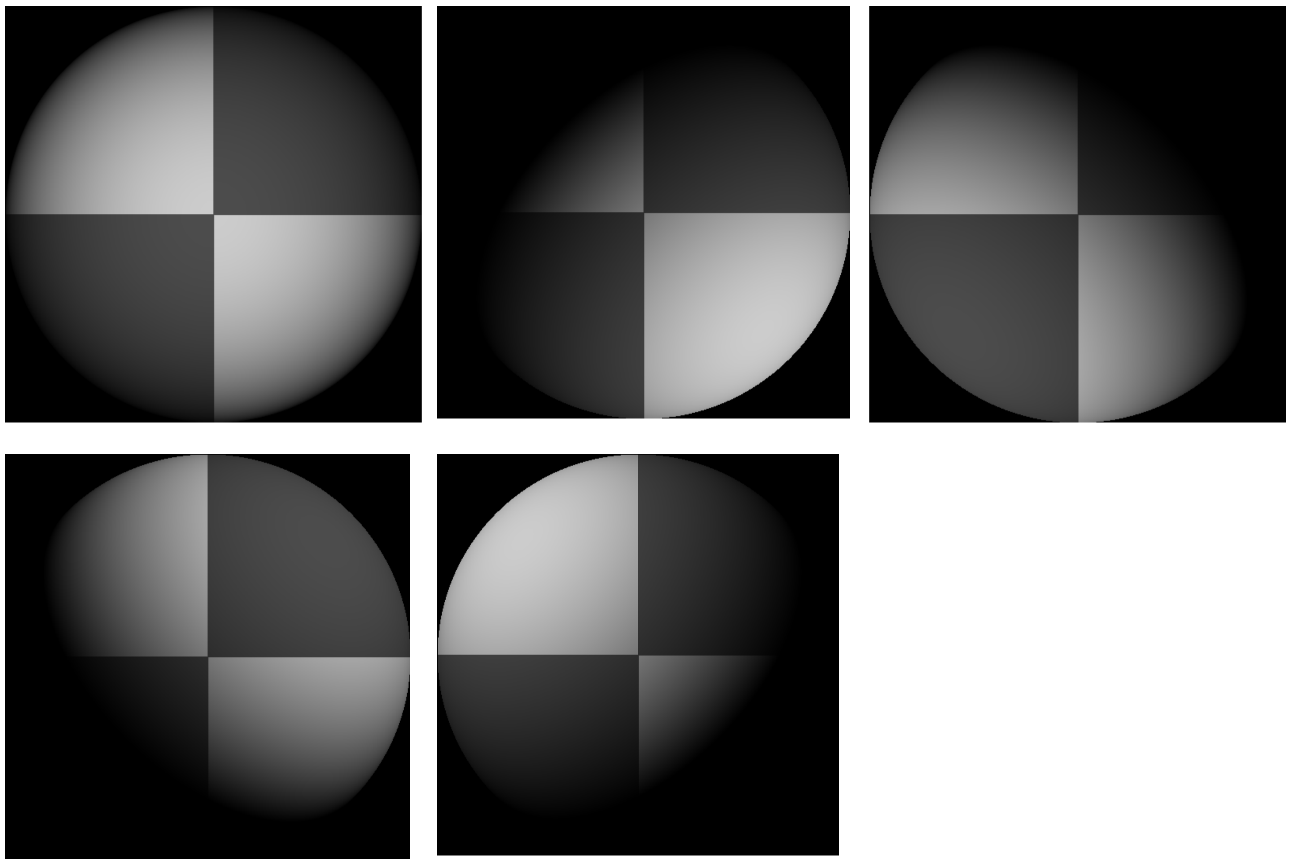
$$\mathtt{S}^+ \mathbf{e}(x,y) = \mathbf{b}(x,y), \text{ where } \mathtt{S}^+ = (\mathtt{S}^\top \mathtt{S})^{-1}\mathtt{S}^\top$$

Same for all pixel coordinates

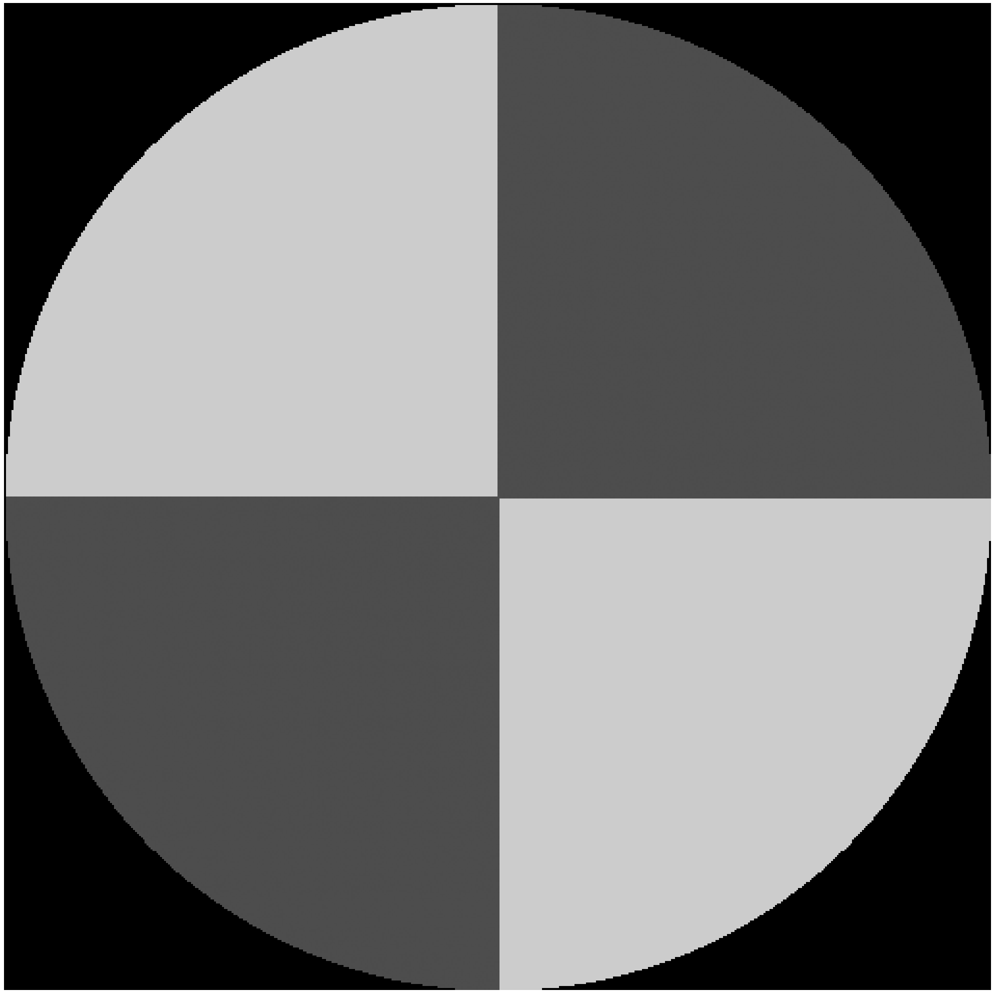
$$\mathbf{b}(x,y) = a(x,y)\hat{\mathbf{n}}(x,y), \text{ where } a(x,y) = \|\mathbf{b}(x,y)\| \text{ and } \hat{\mathbf{n}}(x,y) = \frac{\mathbf{b}(x,y)}{\|\mathbf{b}(x,y)\|}$$

albedo  + surface normals

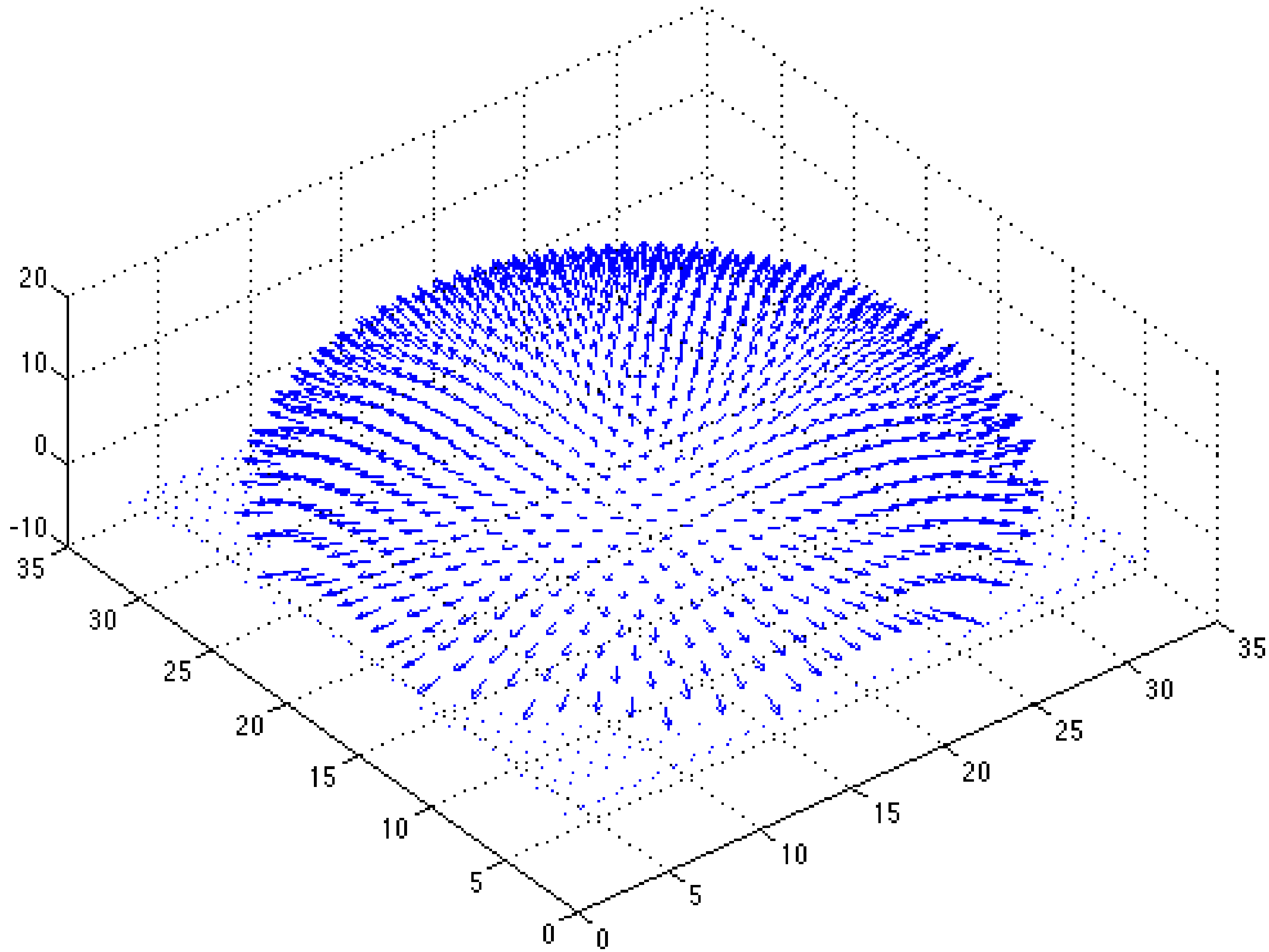
Input Images



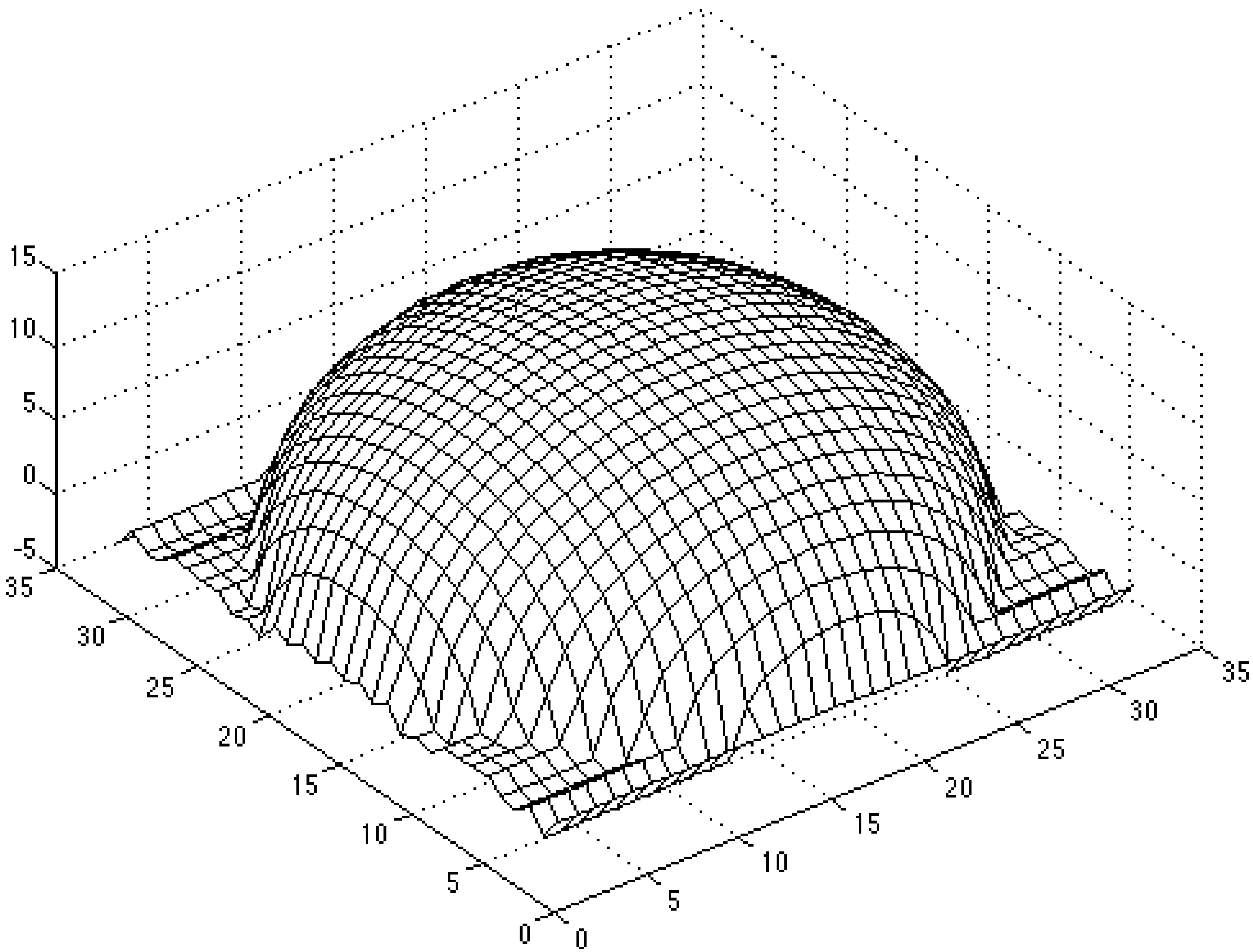
Recovered albedo



Recovered normal field

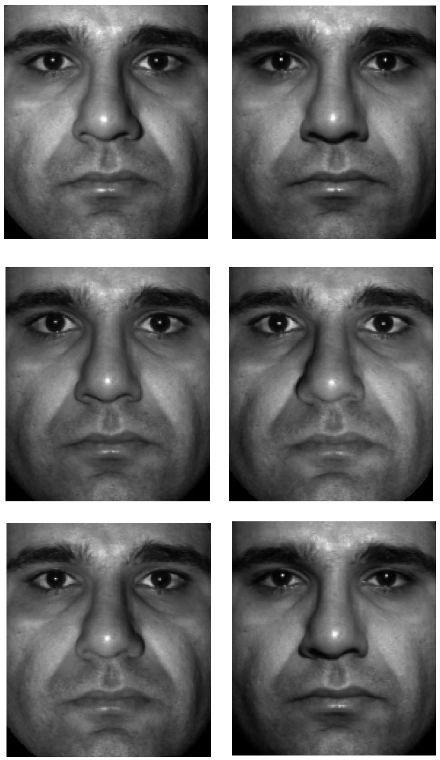


Surface recovered by integration

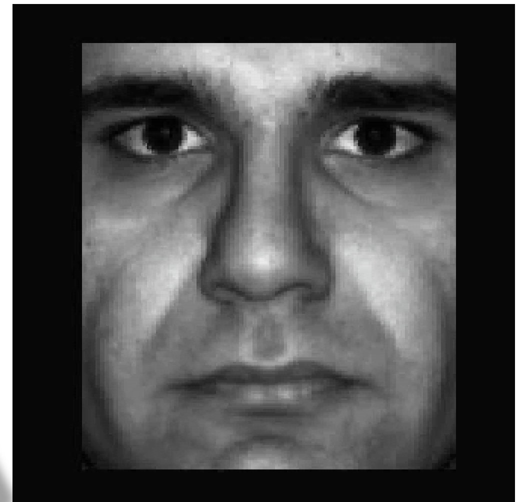


Use a convex mask to only include object

An example of photometric stereo



Images with known associated light sources



Albedo

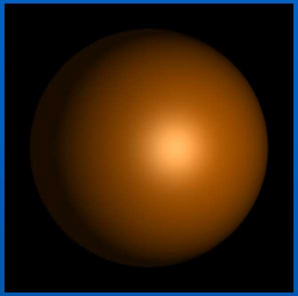
Surface (from normals)

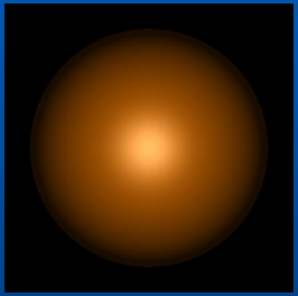
Surface (albedo texture map)

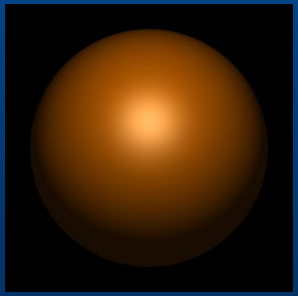
Use a convex mask to only include object

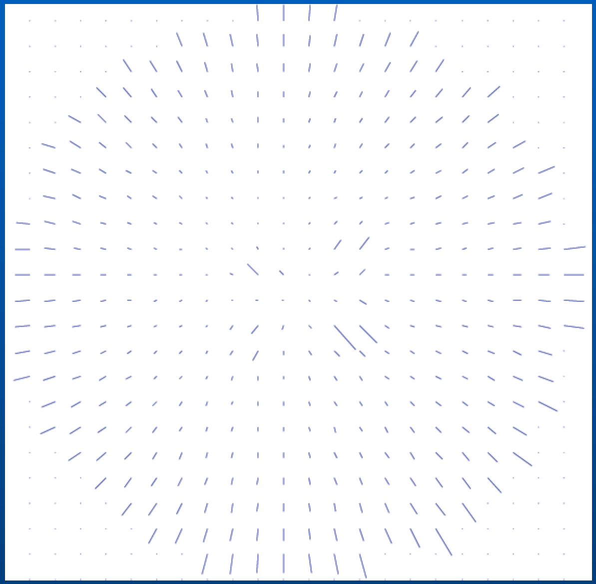
SUV Color Space for Photometric Stereo of Glossy Objects

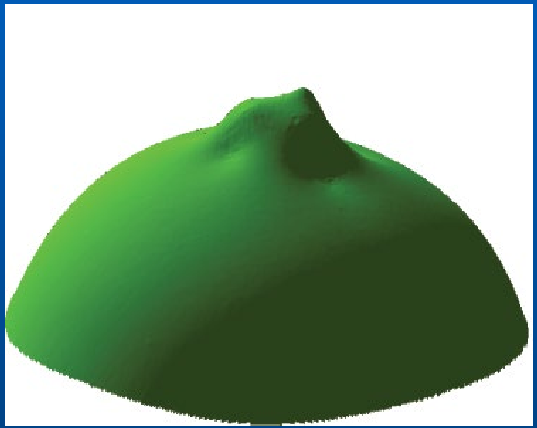
Motivation: Lambertian Algorithm Applied to non-Lambertian Surface: Photometric Stereo





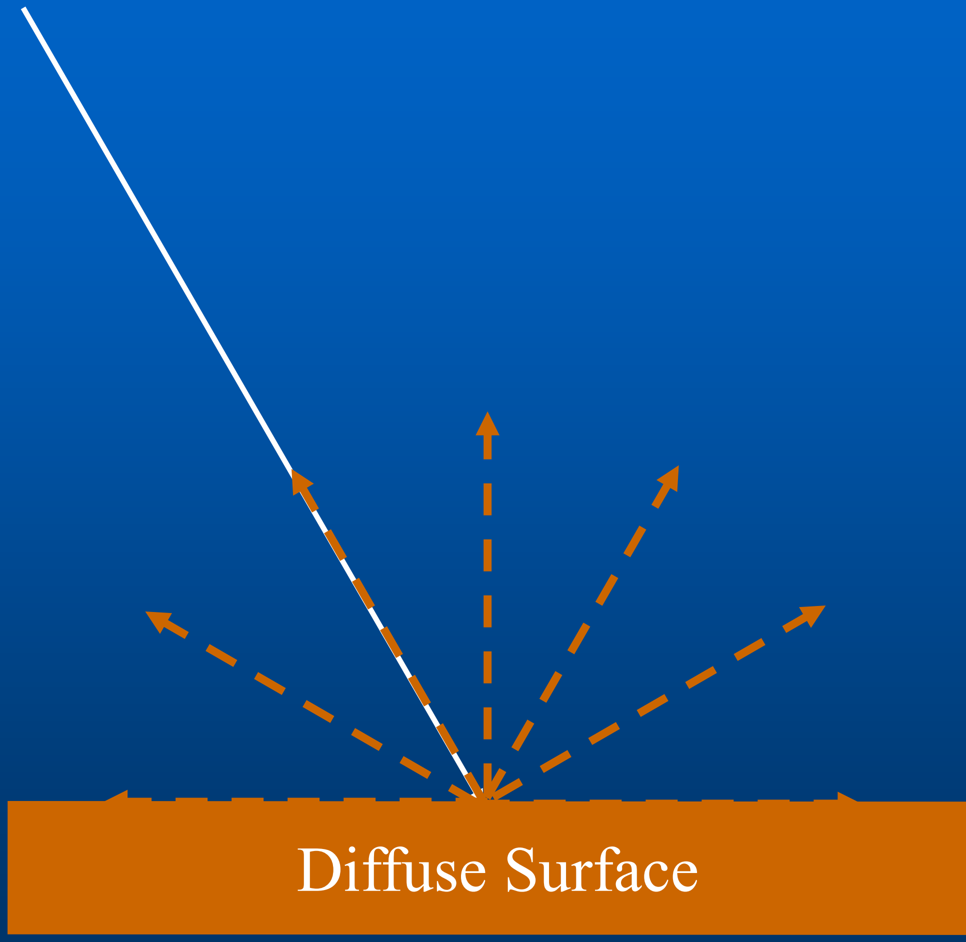


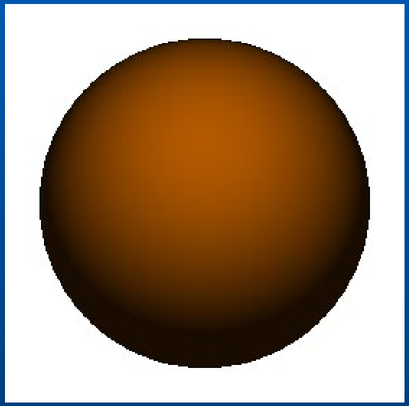




Use a convex mask to only include object

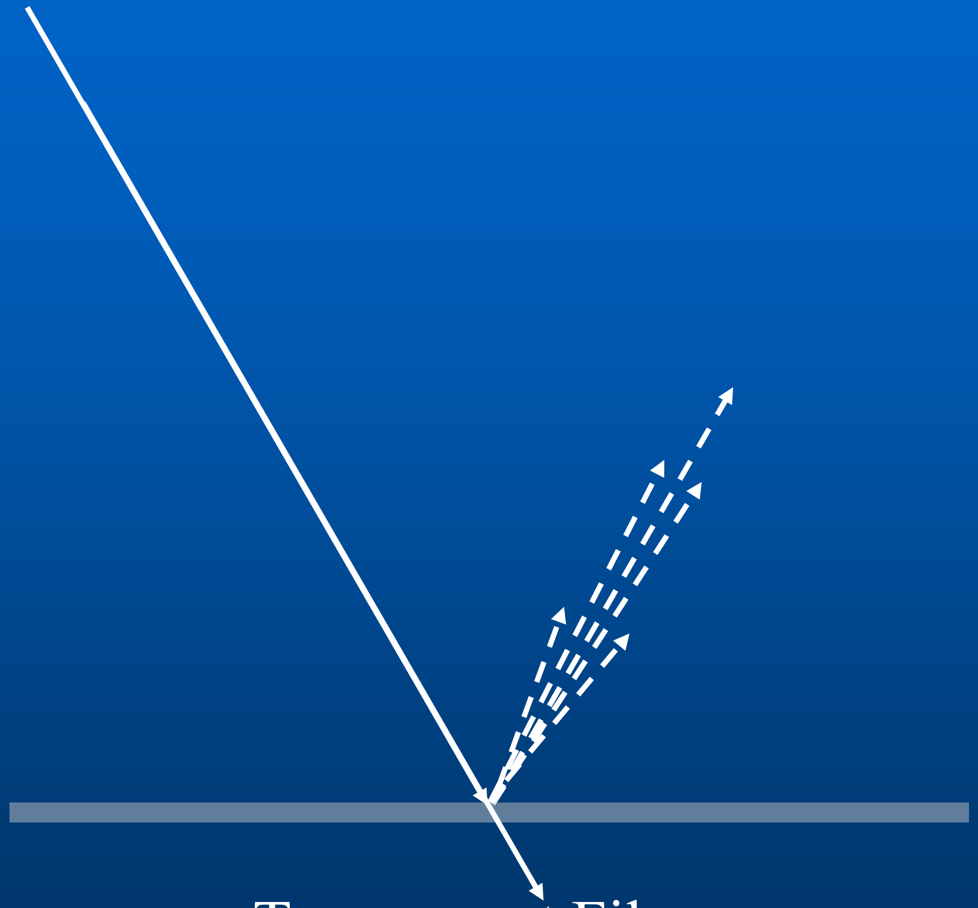
Dichromatic Reflection Model

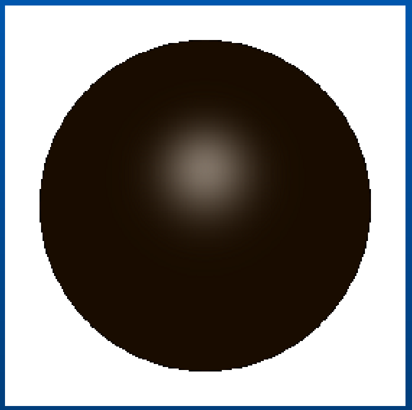




Color depends on light source color and diffuse color

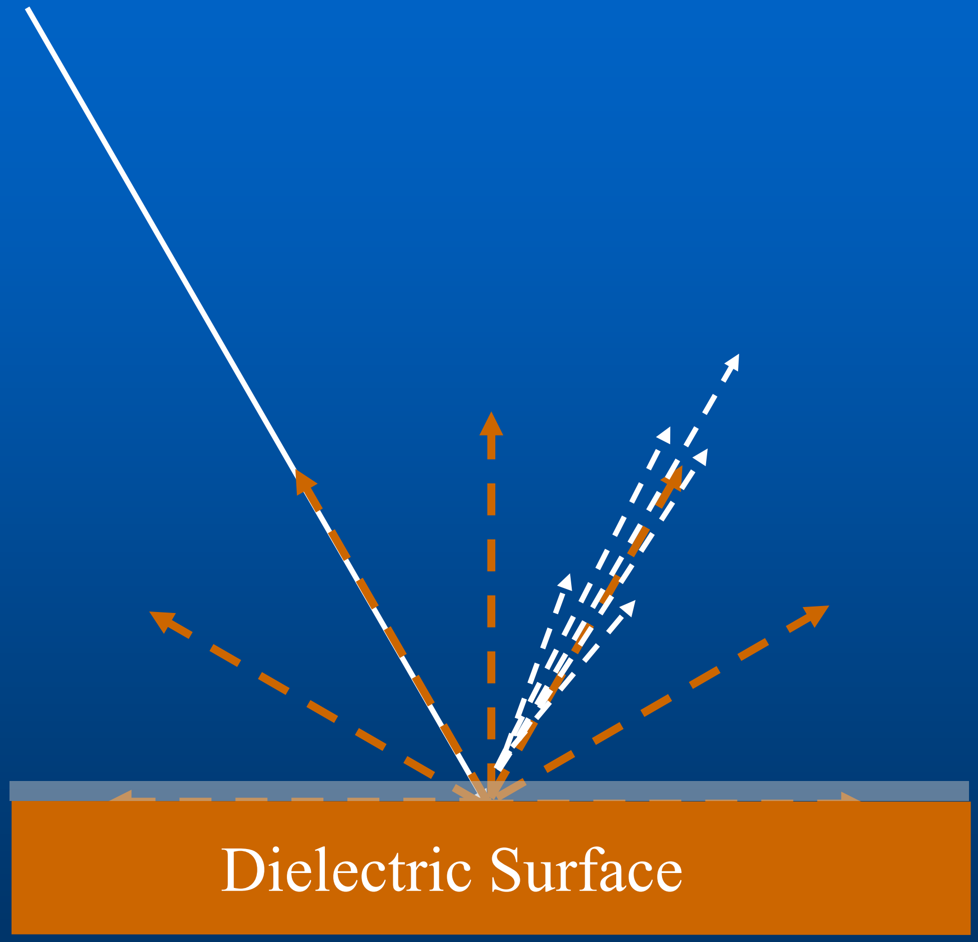
Dichromatic Reflection Model





Color of light source

Dichromatic Reflection Model



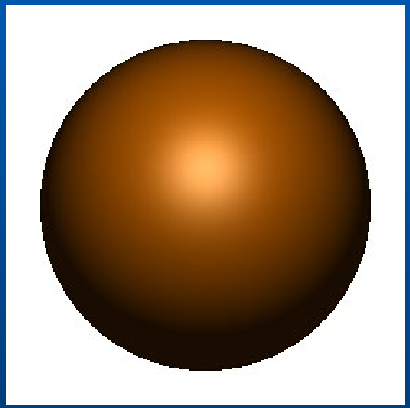


Image formation: Color Channel k

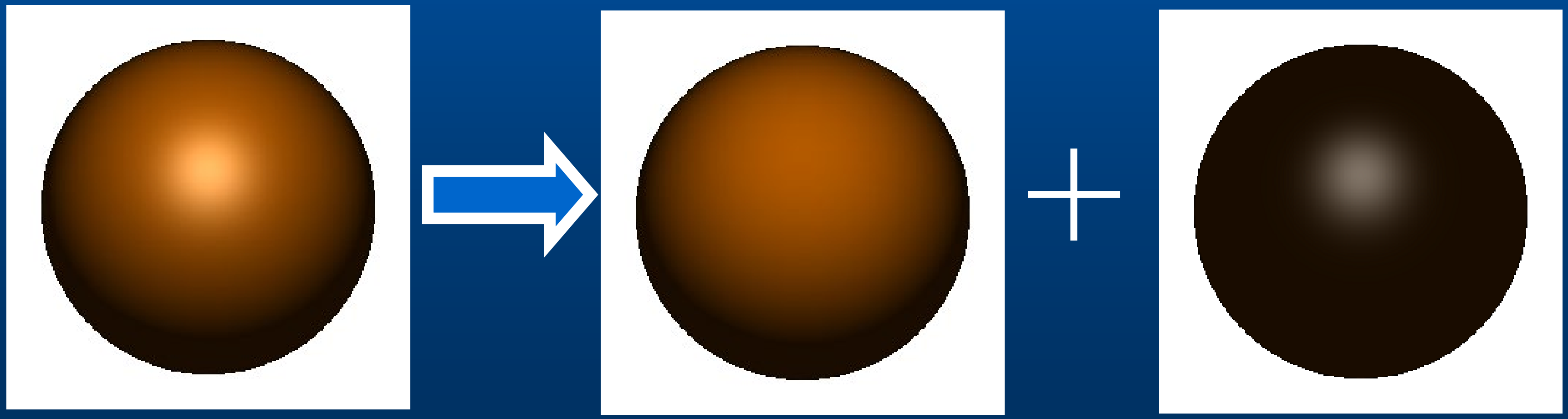
$$L(\lambda) = \text{Spectral Power Distribution of light source}$$

$$C_k(\lambda) = \text{Camera Sensitivity}$$

$$S_k = \int C_k(\lambda) L(\lambda) d\lambda. \quad \text{Specular Color}$$

$$D_k = \int C_k(\lambda) L(\lambda) g_d(\lambda) d\lambda \quad \text{Diffuse Color}$$

$$I_k = \left(D_k f_d + S_k f_s(\boldsymbol{\theta})\right) \hat{\mathbf{n}} \cdot \hat{\mathbf{l}}, \qquad \boldsymbol{\theta} = (\theta_i, \phi_i, \theta_r, \phi_r)$$



Where f_d and f_s are the diffuse and specular BRDF

Image formation: 3 color channels

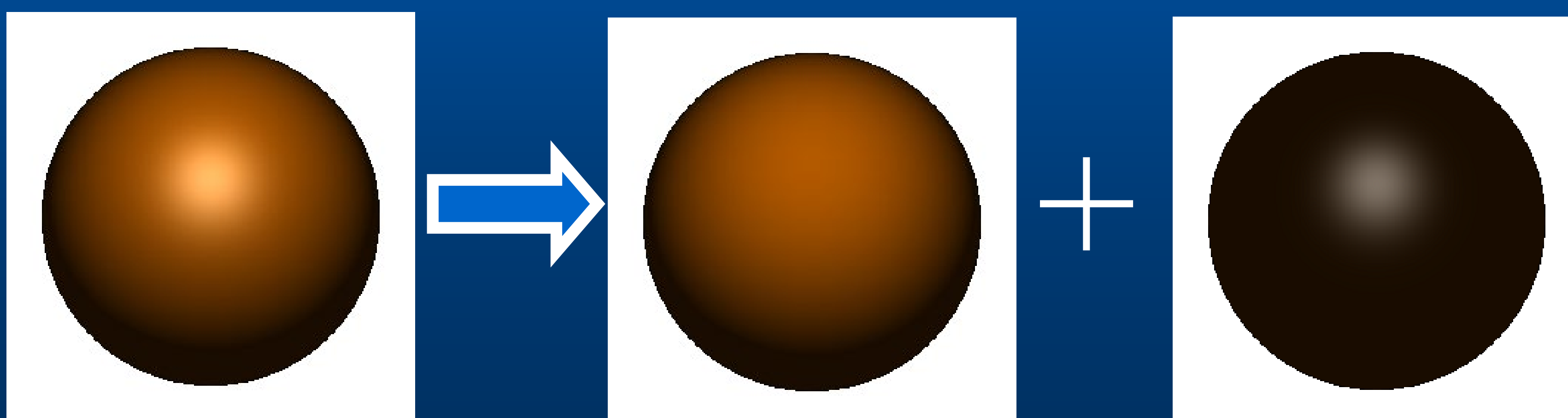
$$I_k = (D_k f_d + S_k f_s(\boldsymbol{\theta}))\, \hat{\mathbf{n}} \cdot \hat{\mathbf{l}},$$

$$\boldsymbol{\theta} = (\theta_i, \phi_i, \theta_r, \phi_r)$$

$$\begin{bmatrix} I_r \\ I_g \\ I_b \end{bmatrix} = \left(f_d \hat{\mathbf{n}} \cdot \hat{l}\right) \begin{bmatrix} D_r \\ D_g \\ D_b \end{bmatrix} + \left(f_s(\theta) \hat{\mathbf{n}} \cdot \hat{l}\right) \begin{bmatrix} S_r \\ S_g \\ S_b \end{bmatrix}$$

Image color lies in span of diffuse color ***D*** and specular color ***S***

$$\mathbf{I} = (f_d \hat{n} \cdot \hat{l})\mathbf{D} + (f_s(\theta) \hat{n} \cdot \hat{l})\mathbf{S}$$

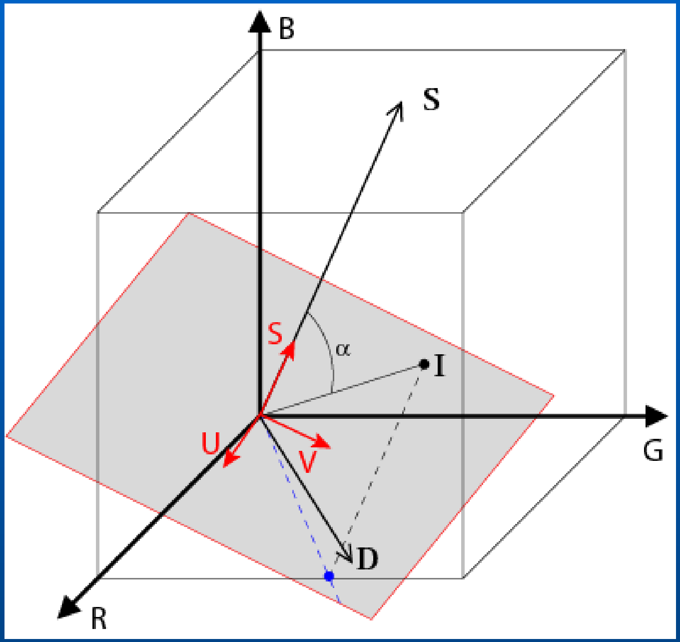


Varying diffuse color

Note:

- Diffuse color D varies over the image
- Specular color is just color of light source

Data-dependent SUV Color Space



$$\mathbf{I}_{SUV} = [R]\mathbf{I}_{RGB}$$

$$[R] \in SO(3)$$

U,V spans a plane orthogonal to **S**

First row of R is specular color S. Other rows are orthogonal to S

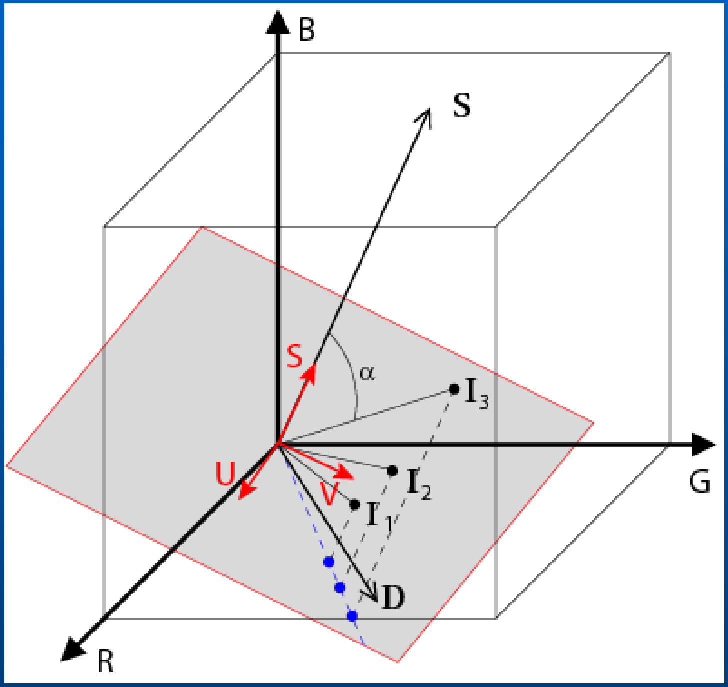
Given color of light source $\mathbf{c} = (R_s, G_s, B_s)^\top$, calculate $\mathtt{R} \in \mathrm{SO}(3)$ such that $(1,0,0)^\top = \mathtt{R}\hat{\mathbf{c}}$

$$\hat{\mathbf{c}} = \frac{\mathbf{c}}{\|\mathbf{c}\|}$$

$$\mathtt{R} = \begin{bmatrix} \hat{\mathbf{c}}^\top \\ [\hat{\mathbf{c}}]^\perp \end{bmatrix}, \text{ where } [\hat{\mathbf{c}}]^\perp \hat{\mathbf{c}} = \mathtt{0} \text{ (i.e., } [\hat{\mathbf{c}}]^\perp \text{ is left null space of } \hat{\mathbf{c}}$$

if $\det(\mathtt{R}) = -1$, then negate last row of $\mathtt{R}$

Properties of SUV



- Data-dependent.
- Rotational (hence, linear) Transformation.
- The S channel encodes the entire specular component and an unknown amount of diffuse component.
- Shading information is preserved in u and v channels.
- Diffuse image

$$D = \sqrt{U^2 + V^2}$$

Example

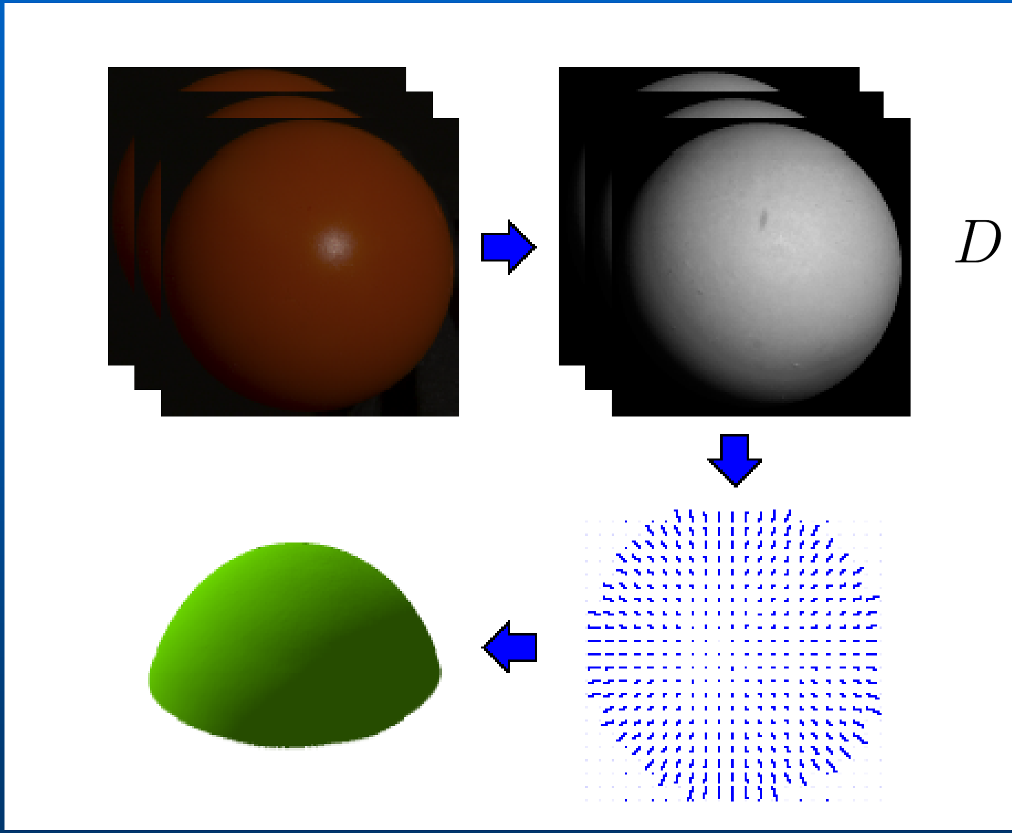
RGB

S channel

U channel

V channel

Multi-channel Photometric Stereo



$$D = \sqrt{U^2 + V^2}$$

Use a convex mask to only include object

Multi-channel Photometric Stereo

$\mathbf{J} = [I_U \quad I_V]^\top$

$\mathbf{J}^k$: 2-channel color vector under the k^{th} light source.

$\hat{\mathbf{l}}^k$: The k^{th}three light source directions.

$\boldsymbol{\rho}$: 2-channel UV albedo.

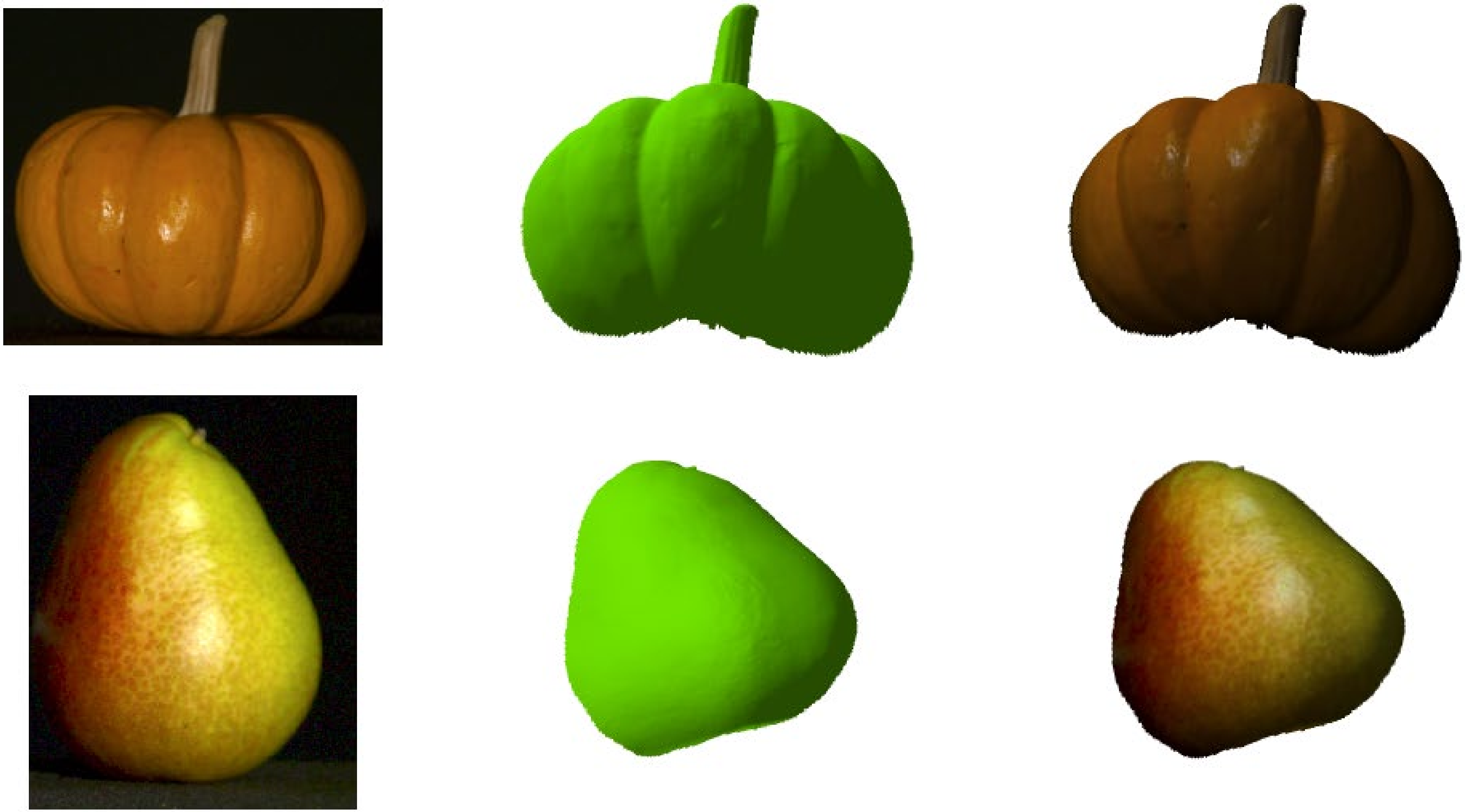
$$\mathbf{J}^k = [I_U^k, I_V^k]^\top = (\hat{\mathbf{n}} \cdot \hat{\mathbf{l}}^k)\boldsymbol{\rho},$$

$$\text{Shading vector} \quad = \mathbf{F} \quad = [f^1, f^2, f^3]^\top \quad = [\hat{\mathbf{l}}^1 \ \hat{\mathbf{l}}^2 \ \hat{\mathbf{l}}^3]^\top \hat{\mathbf{n}}$$

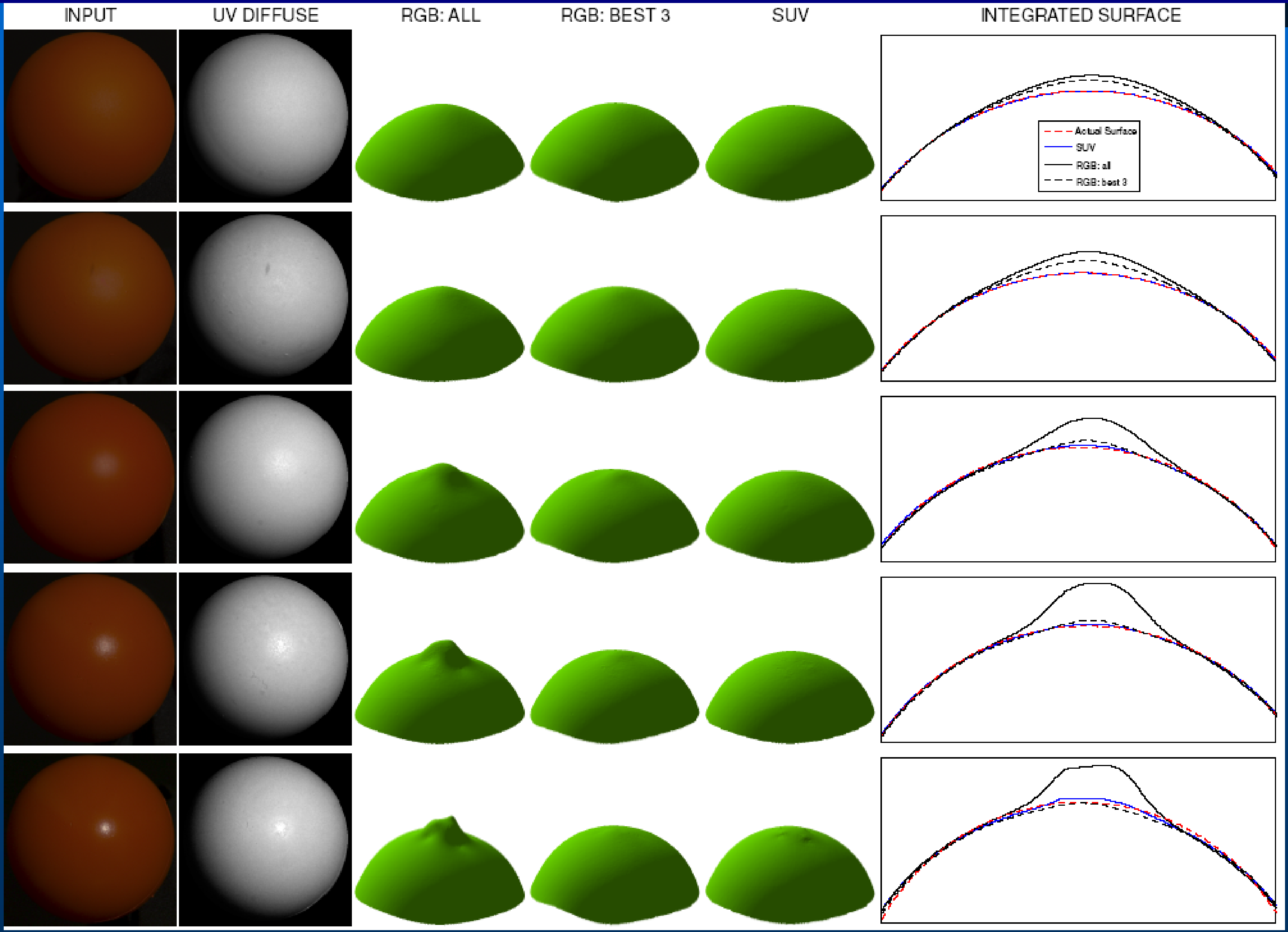
$$\text{Intensity matrix} \quad = [J] \quad = \begin{bmatrix} J_1^1 & J_2^1 \\ J_1^2 & J_2^2 \\ J_1^3 & J_2^3 \end{bmatrix} \quad = \begin{bmatrix} f^1 \rho_U & f^1 \rho_V \\ f^2 \rho_U & f^2 \rho_V \\ f^3 \rho_U & f^3 \rho_V \end{bmatrix} \quad = \mathbf{F}\boldsymbol{\rho}^\top.$$

The least squares estimate of the shading vector $\mathbf{F}$ is the principal eigenvector of $[J][J]^\top$. Once the shading vector is known, the surface normal is found by solving the matrix equation $\mathbf{F} = [\hat{\mathbf{l}}^1 \ \hat{\mathbf{l}}^2 \ \hat{\mathbf{l}}^3]^\top \hat{\mathbf{n}}$.

Qualitative Results

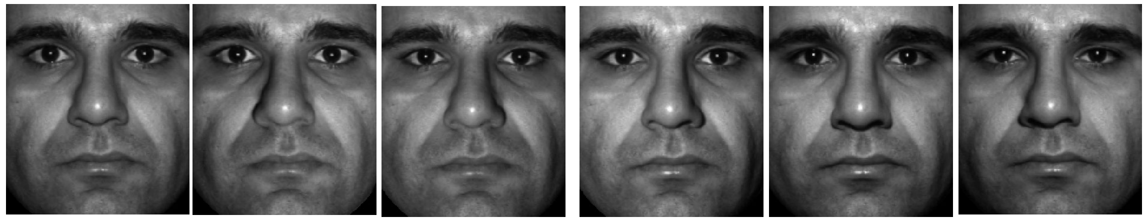


Quantitative Results



3. Photometric Stereo with unknown lighting and Lambertian surfaces

Uncalibrated Photometric Stereo



- For calibrated photometric stereo, we estimated the n by 3 matrix **B** of surface normals scaled by albedo using lighting.
- Uncalibrated Input: Only images. No lighting info.
- Without shadowing, all images lie in 3D subspace of the n-pixel image space spanned by columns of an *n by 3* matrix **B***.
- From 3 or more images, SVD can be used to estimate **B***.
- The n by 3 matrix **B** of surface normals scaled by albedo differs from **B*** by a 3x3 linear transformation **B=AB***.
- After enforcing integrability, one can only estimate shape and albedo (**B**) up to a Generalized Bas Relief (GBR) transformation which has 3 parameters (depth scaling, tilt)

Next Lecture

- Image filtering