Camera Calibration & Radiometry

- **Reading:**
  - –Chapter 2, and sections 3.2, 5.4, Forsyth & Ponce
  - –Chapter 10, Horn
- **Optional reading:**
  - –Chapter 4, Forsyth & Ponce
- **Handouts:** Revised Problem Set 1

February 12, 2008

Plan

- **First part:** how positions in the image relate to 3 d positions in the world.
- **Second part:** how image intensities relate to surface and lighting properties in the world.

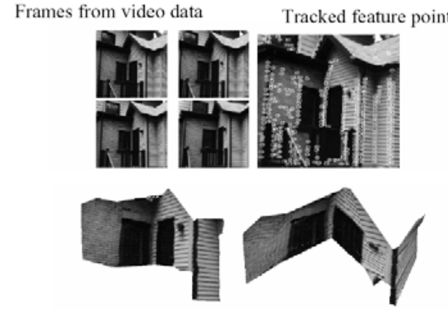
Last Lecture:

Camera calibration

- Use the camera to tell you things about the world.
  - – Relationship between coordinates in the world and coordinates in the image: geometric camera calibration.
  - – (Later we'll discuss relationship between intensities in the world and intensities in the image: photometric camera calibration.)

Motivation for camera calibration:

relating image measurements to positions out in the world



Inferred 3-d shape of building

Combining extrinsic and intrinsic calibration parameters

$$\vec{p} = \frac{1}{z}\begin{pmatrix} K & \vec{0} \end{pmatrix} {}^{C}\vec{P} \qquad \text{Intrinsic}$$

$$\begin{pmatrix} {}^{C}\vec{P} \end{pmatrix} = \begin{pmatrix} {}^{C}_{W}R & {}^{C}O_{W} \\ 0\ 0\ 0 & 1 \end{pmatrix} \begin{pmatrix} \vec{P} \end{pmatrix} \qquad \text{Extrinsic}$$

$$\vec{p} = \frac{1}{z} K \begin{pmatrix} {}^{C}_{W}R & {}^{C}O_{W} \end{pmatrix} \vec{P}$$

$$\vec{p} = \frac{1}{z} M \vec{P}$$

Forsyth&Ponce

Other ways to write the same equation

pixel coordinates — world coordinates

$$\vec{p} = \frac{1}{z} M\vec{P}$$

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \frac{1}{z}\begin{pmatrix} . & m_1^T & . & . \\ . & m_2^T & . & . \\ . & m_3^T & . & . \end{pmatrix} \begin{pmatrix} {}^{W}x \\ {}^{W}y \\ {}^{W}z \\ 1 \end{pmatrix}$$

$$u = \frac{m_1 \cdot \vec{P}}{m_3 \cdot \vec{P}} \qquad v = \frac{m_2 \cdot \vec{P}}{m_3 \cdot \vec{P}}$$

z is in the *camera* coordinate system, but we can solve for that, since $1 = \frac{m_3 \cdot \vec{P}}{z}$, leading to:

Calibration target

The Opti-CAL Calibration Target Image

http://www.kinetic.bc.ca/CompVision/opti-CAL.html

Camera calibration

From before, we had these equations relating image positions, u,v, to points at 3-d positions P (in homogeneous coordinates):

$$u = \frac{m_1 \cdot \vec{P}}{m_3 \cdot \vec{P}}$$

$$v = \frac{m_2 \cdot \vec{P}}{m_3 \cdot \vec{P}}$$

So for each feature point, i, we have:

$$(m_1 - u_i m_3) \cdot \vec{P}_i = 0$$

$$(m_2 - v_i m_3) \cdot \vec{P}_i = 0$$

Camera calibration

Stack all these measurements of i=1…n points

$$(m_1 - u_i m_3) \cdot \vec{P}_i = 0$$

$$(m_2 - v_i m_3) \cdot \vec{P}_i = 0$$

into a big matrix:

$$\begin{pmatrix} P_1^T & 0^T & -u_1 P_1^T \\ 0^T & P_1^T & -v_1 P_1^T \\ \cdots & \cdots & \cdots \\ P_n^T & 0^T & -u_n P_n^T \\ 0^T & P_n^T & -v_n P_n^T \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Camera calibration

In vector form:

$$\begin{pmatrix} P_1^T & 0^T & -u_1 P_1^T \\ 0^T & P_1^T & -v_1 P_1^T \\ \cdots & \cdots & \cdots \\ P_n^T & 0^T & -u_n P_n^T \\ 0^T & P_n^T & -v_n P_n^T \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Showing all the elements:

$$\begin{pmatrix} P_{1x} & P_{1y} & P_{1z} & 1 & 0 & 0 & 0 & 0 & -u_1 P_{1x} & -u_1 P_{1y} & -u_1 P_{1z} & -u_1 \\ 0 & 0 & 0 & 0 & P_{1x} & P_{1y} & P_{1z} & 1 & -v_1 P_{1x} & -v_1 P_{1y} & -v_1 P_{1z} & -v_1 \\ & & & & & & \cdots & \cdots & \cdots & & & \\ P_{nx} & P_{ny} & P_{nz} & 1 & 0 & 0 & 0 & 0 & -u_n P_{nx} & -u_n P_{ny} & -u_n P_{nz} & -u_n \\ 0 & 0 & 0 & 0 & P_{nx} & P_{ny} & P_{nz} & 1 & -v_n P_{nx} & -v_n P_{ny} & -v_n P_{nz} & -v_n \end{pmatrix} \begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Camera calibration

$$\begin{pmatrix} P_{1x} & P_{1y} & P_{1z} & 1 & 0 & 0 & 0 & 0 & -u_1 P_{1x} & -u_1 P_{1y} & -u_1 P_{1z} & -u_1 \\ 0 & 0 & 0 & 0 & P_{1x} & P_{1y} & P_{1z} & 1 & -v_1 P_{1x} & -v_1 P_{1y} & -v_1 P_{1z} & -v_1 \\ & & & & & & \cdots & \cdots & \cdots & & & \\ P_{nx} & P_{ny} & P_{nz} & 1 & 0 & 0 & 0 & 0 & -u_n P_{nx} & -u_n P_{ny} & -u_n P_{nz} & -u_n \\ 0 & 0 & 0 & 0 & P_{nx} & P_{ny} & P_{nz} & 1 & -v_n P_{nx} & -v_n P_{ny} & -v_n P_{nz} & -v_n \end{pmatrix} \begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Q m = 0

We want to solve for the unit vector m (the stacked one) that minimizes $|Qm|^2$

The minimum eigenvector of the matrix $Q^T Q$ gives us that (see Forsyth&Ponce, 3.1)

Camera calibration

Once you have the M matrix, can recover the intrinsic and extrinsic parameters as in Forsyth&Ponce, sect. 3.2.2.

$$\mathcal{M} = \begin{pmatrix} \alpha \boldsymbol{r}_1^T - \alpha \cot\theta \boldsymbol{r}_2^T + u_0 \boldsymbol{r}_3^T & \alpha t_x - \alpha \cot\theta t_y + u_0 t_z \\ \frac{\beta}{\sin\theta} \boldsymbol{r}_2^T + v_0 \boldsymbol{r}_3^T & \frac{\beta}{\sin\theta} t_y + v_0 t_z \\ \boldsymbol{r}_3^T & t_z \end{pmatrix}$$

What makes a valid M matrix?

A projection matrix can be written explicitly as a function of its five intrinsic parameters (α, β, u_0, v_0, and θ) and its six extrinsic ones (the three angles defining $\mathcal{R}$ and the three coordinates of $\boldsymbol{t}$), namely,

$$\mathcal{M} = \begin{pmatrix} \alpha \boldsymbol{r}_1^T - \alpha \cot\theta \boldsymbol{r}_2^T + u_0 \boldsymbol{r}_3^T & \alpha t_x - \alpha \cot\theta t_y + u_0 t_z \\ \frac{\beta}{\sin\theta} \boldsymbol{r}_2^T + v_0 \boldsymbol{r}_3^T & \frac{\beta}{\sin\theta} t_y + v_0 t_z \\ \boldsymbol{r}_3^T & t_z \end{pmatrix}, \qquad (2.17)$$

where $\boldsymbol{r}_1^T$, $\boldsymbol{r}_2^T$, and $\boldsymbol{r}_3^T$ denote the three rows of the matrix $\mathcal{R}$ and t_x, t_y, and t_z are the coordinates of the vector $\boldsymbol{t}$.

- $M = (\mathrm{A} \quad \vec{\mathrm{b}}) = \begin{pmatrix} \vec{a}_1^T & \\ \vec{a}_2^T & \vec{b} \\ \vec{a}_3^T & \end{pmatrix}$ defined only up to a scale;

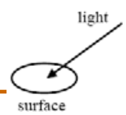
  normalize M so that $|\vec{a}_3^T| = |\vec{r}_3^T| = 1$.

- M is a perspective projection matrix iff $Det(A) \neq 0$

Today's class

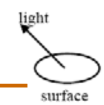
- First part: how *positions* in the image relate to 3-d positions in the world.
- Second part: how the *intensities* in the image relate surface and lighting properties in the world.

Irradiance, E



- Light power per unit area (watts per square meter) incident on a surface.
- The units tell you what to integrate over to find the energy impinging on a given area.
- E times pixel area, times exposure time gives the pixel intensity out (for linear sensor response)

Radiance, L

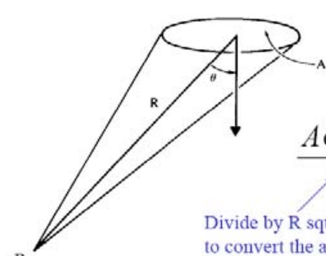


- Amount of light radiated from a surface into a given solid angle per unit area (watts per square meter per steradian).
- Note: the area is the foreshortened area, as seen from the direction that the light is being emitted.
- Informally, radiance tells you the "brightness".

Solid angle

- The solid angle subtended by a cone of rays is the area of a unit sphere (centered at the cone origin) intersected by the cone.
- All possible angles from a point covers 4π steradians.
- A hemisphere covers 2π steradians, etc.

What's the solid angle subtended by this patch, area A, seen from P?



Multiply by cos(θ) to account for foreshortening

$$\frac{A\cos(\theta)}{R^2}$$

Divide by R squared to convert the area to what you'd see on a unit sphere

Image irradiance/scene radiance relationship

- The definition of scene radiance is constructed so that image irradiance is proportional to scene radiance.

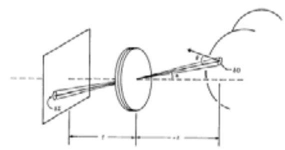


Figure 10-4. To see how image irradiance is related to the radiance of the surface, we must determine the size of the region in the image that corresponds to the patch on the surface.

Horn, sect. 10.3

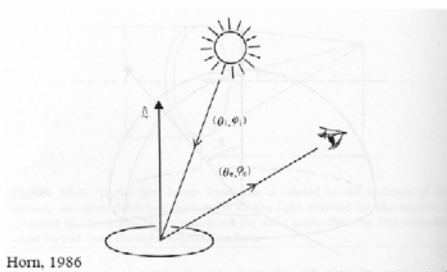
Scene radiance

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

Image irradiance

How the brightness depends on the surface properties: BRDF

- Bidirectional reflectance distribution function tells how bright a surface appears when viewed from one direction while light falls on it from another.

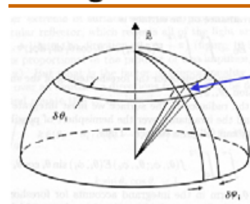


Horn, 1986

Figure 10-7. The bidirectional reflectance distribution function is the ratio of the radiance of the surface patch as viewed from the direction (θ_e, ϕ_e) to the irradiance resulting from illumination from the direction (θ_i, ϕ_i).

$$BRDF = f(\theta_i, \phi_i, \theta_e, \phi_e) = \frac{L(\theta_e, \phi_e)}{E(\theta_i, \phi_i)}$$

How does the world give us the brightness we observe at a point?



Solid angle of this patch:

$$\delta\omega = \sin(\theta_i)\, \delta\theta_i\, \delta\phi_i$$

Let radiance per solid angle be:

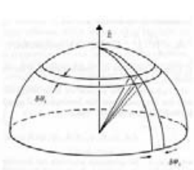
$$E(\theta_i, \phi_i)$$

Integrate all the source radiance impinging on the surface

The the radiance from this patch toward the origin is:

$$E(\theta_i, \phi_i) \sin(\theta_i)\, \delta\theta_i\, \delta\phi_i$$

Accounting for extended light sources



The total irradiance of the surface is:

$$E_0 = \int_{-\pi}^{\pi} \int_{0}^{\pi/2} E(\theta_i, \phi_i) \sin(\theta_i) \cos(\theta_i)\, d\theta_i\, d\phi_i$$

Accounting for the foreshortened area of center patch relative to illuminant.

The total radiance reflected from the surface patch is:

$$L(\theta_e, \phi_e) = \int_{-\pi}^{\pi} \int_{0}^{\pi/2} f(\theta_i, \phi_i, \theta_e, \phi_e)\, E(\theta_i, \phi_i) \sin(\theta_i) \cos(\theta_i)\, d\theta_i\, d\phi_i$$

What you'd like to pull out from L

Pixel intensities may be proportional to radiance reflected from the surface patch:

$$L(\theta_e, \phi_e) = \int_{-\pi}^{\pi} \int_{0}^{\pi/2} f(\theta_i, \phi_i, \theta_e, \phi_e)\, E(\theta_i, \phi_i) \sin(\theta_i) \cos(\theta_i)\, d\theta_i\, d\phi_i$$

θ_e, ϕ_e **surface orientation relative to camera**

$f(\theta_i, \phi_i, \theta_e, \phi_e)$ **surface BRDF**

$E(\theta_i, \phi_i)$ **illumination conditions**

θ_i, ϕ_i **surface orientation relative to illumination**

That's hard, so let's focus on special cases for the rest of this lecture.

Special case BRDF: Lambertian reflectance

BRDF is a constant. These surfaces look equally bright from all viewing directions.

$$f(\theta_i,\phi_i,\theta_e,\phi_e)=\frac{1}{\pi}$$

Radiance reflected from Lambertian surface illuminated by point source located at (θ_s,ϕ_s)

$$L(\theta_e,\phi_e)=\int_{-\pi}^{\pi}\int_{0}^{\pi/2}\frac{1}{\pi}\underbrace{E\frac{\delta(\theta_i-\theta_s)\delta(\phi_i-\phi_s)}{\sin(\theta_i)}}_{E(\theta_i,\phi_i)}\sin(\theta_i)\cos(\theta_i)\,d\theta_i\,d\phi_i$$

Special case BRDF: Lambertian reflectance

BRDF is a constant. These surfaces look equally bright from all viewing directions.

$$f(\theta_i,\phi_i,\theta_e,\phi_e)=\frac{1}{\pi}$$

Radiance reflected from Lambertian surface illuminated by point source located at (θ_s,ϕ_s)

$$L(\theta_e,\phi_e)=\int_{-\pi}^{\pi}\int_{0}^{\pi/2}\frac{1}{\pi}E\frac{\delta(\theta_i-\theta_s)\delta(\phi_i-\phi_s)}{\sin(\theta_i)}\sin(\theta_i)\cos(\theta_i)\,d\theta_i\,d\phi_i$$

$$=\tfrac{1}{\pi}E\cos(\theta_s)\approx\cos(\theta_s)$$

Special case BRDF: Specular Reflection

- Surface looks bright only when the viewing angles equal the illumination angles

$$f(\theta_i,\phi_i,\theta_e,\phi_e)=\frac{\delta(\theta_e-\theta_i)\delta(\phi_e-\phi_i-\pi)}{\sin(\theta_i)\cos(\theta_i)}$$ Horn 10.6

- Radiance reflected from a specular surface illuminated by point light source locates at (θ_s,ϕ_s)

$$L(\theta_e,\phi_e)=\int_{-\pi}^{\pi}\int_{0}^{\pi/2}\frac{\delta(\theta_e-\theta_i)\delta(\phi_e-\phi_i-\pi)}{\sin(\theta_i)\cos(\theta_i)}E(\theta_i,\phi_i)\sin(\theta_i)\cos(\theta_i)\,d\theta_i\,d\phi_i$$

$$=E(\theta_e,\phi_e-\pi)$$

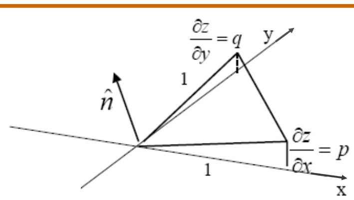
Reflectance map

- For orthographic projection, and light sources at infinity, the reflectance map is a useful tool for describing the relationship of surface orientation to image intensity.
- Describes the image intensity for a given surface orientation.
- Parameterize surface orientation by the partial derivatives p and q of surface height z.

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

- Useful in recovering surface shape from images

Relate surface normal to p & q



$\mathbf{r}_x=[1,0,p]^T$ $\mathbf{r}_y=[0,1,q]^T$ $\mathbf{n}=\mathbf{r}_x\times\mathbf{r}_y=[-p,-q,1]^T$

Unit normal to surface:

$$\hat{n}=\frac{\vec{n}}{|\vec{n}|}=\frac{(-p\quad -q\quad 1)^T}{\sqrt{1+p^2+q^2}}$$

Refl map for point source (in direction $\hat{s}$) Lambertian surface

Unit vector to source:

$$\hat{s}=\frac{(-p_s\quad -q_s\quad 1)^T}{\sqrt{1+p_s^{\,2}+q_s^{\,2}}}$$

For a Lambertian surface, $\overbrace{L(\theta_e,\phi_e)=\tfrac{1}{\pi}E\cos(\theta_s)\approx\cos(\theta_s)}^{\text{Scene Radiance}}$

$$\cos(\theta_s)=\hat{\mathbf{n}}\cdot\hat{\mathbf{s}}$$

$$=k\frac{1+p_sp+q_sq}{\sqrt{1+p^2+q^2}\sqrt{1+p_s^2+q_s^2}}\quad\approx R(p,q)$$

Picture of Lambertian refl map

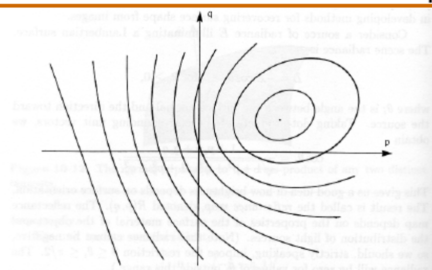
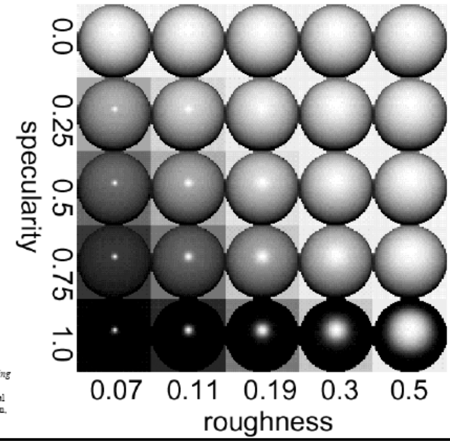


Figure 10-13. The reflectance map is a plot of brightness as a function of surface orientation. Here it is shown as a contour map in gradient space. In the case of a Lambertian surface under point-source illumination, the contours turn out to be nested conic sections. The maximum of $R(p,q)$ occurs at the point $(p,q) = (p_s, q_s)$, found inside the nested conic sections, while $R(p,q) = 0$ all along the line on the left side of the contour map.

Horn, 1986



W. T. Freeman, *Exploiting the generic viewpoint assumption*, International Journal Computer Vision, 20 (3), 243-261, 1996

What constraints are there for form of reflectance map?

– Radiance cannot be negative

– Defined only for $0 \le \theta_i \le \pi/2$

For values outside this range, the radiance is zero.

– Maximum value is 1.

Linear Reflectance Map

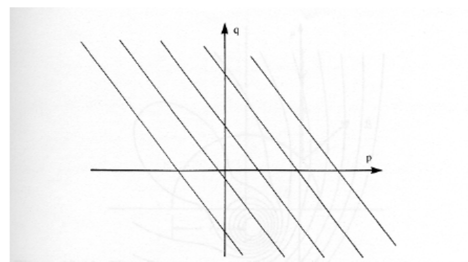


Figure 10-14. In the case of the material in the maria of the moon, the reflectance map can be closely approximated by a function of a linear combination of the components of the gradient. The contours of constant brightness are parallel straight lines in gradient space.

Horn, 1986

Linear shading: 1st order terms of Lambertian shading

Lambertian surface illuminated by point source

$$R(p,q) = k\frac{1 + p_s p + q_s q}{\sqrt{1+p^2+q^2}\sqrt{1+p_s^2+q_s^2}}$$

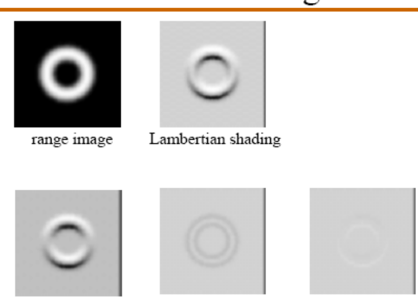
1st order Taylor series about p=q=0

$$\approx k_2 + \left.\frac{\partial R(p,q)}{\partial p}\right|_{p=0,q=0} p + \left.\frac{\partial R(p,q)}{\partial q}\right|_{p=0,q=0} q$$

$$= k_2(1 + p_s p + q_s q)$$

See Pentland, IJCV vol. 1 no. 4, 1990.

Linear shading



Pentland 1990, Adelson&Freeman, 1991

Advantages of linear shading

- Linear relationship between surface range map and rendered image.
- Rendering is easy: differentiate with respect to azimuthal light source direction.
- Applies: linear sources, or shallow illumination angles and Lambertian surface.
- Allows for very simple inverse transformation from rendered image to surface range map, which we'll discuss later with shape-from-shading material.

Knowing the reflectance map, can we infer the gradient at any point?

- There is a unique mapping from surface orientation, *(p,q),* to radiance given by the reflectance map
- Inverse mapping is not unique
- An infinite number of orientations give rise to the same brightness
- Brightness has one degree of freedom, orientation has two
- To recover two unknows, we need two equations – two images taken with different lighting will result into two equations

Generic reflectance map

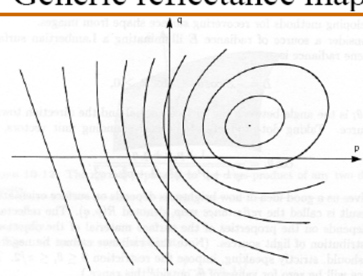
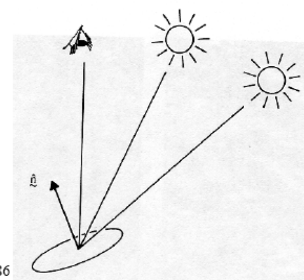


Figure 10-13. The reflectance map is a plot of brightness as a function of surface orientation. Here it is shown as a contour map in gradient space. In the case of a Lambertian surface under point-source illumination, the contours turn out to be nested conic sections. The maximum of $R(p,q)$ occurs at the point $(p,q) = (p_s, q_s)$, found inside the nested conic sections, while $R(p,q) = 0$ all along the line on the left side of the contour map.

Horn, 1986

Photometric stereo



Horn, 1986

Fixed camera and object positions.
Take two or more images under different lighting conditions.

frame 2

frame 11

. . .

. . .

Approach 1

- Photograph the object with a calibration object in the picture, or available in another photograph.
- Use the multiple responses of the calibration object to the different light sources to form a look-up table.
- Index into that table using the multiple responses of the unknown object.
- Handles arbitrary BRDF.

Approach 2

- Assume a particular functional form for the BRDF (Lambertian). Assume known light source positions (point sources at infinity as specified locations).
- Analytically determine the surface slope for each location's collection of image intensities.

Photometric stereo

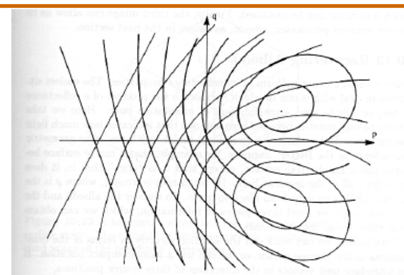


Figure 10-21. In the case of a Lambertian surface illuminated successively by two different point sources, there are at most two surface orientations that produce a particular pair of brightness values. These are found at the intersection of the corresponding contours in two superimposed reflectance maps.

See Forsyth&Ponce sect. 5.4 for procedure. In HW: don't need to integrate the surface normals to get the shape.

From the image under the i[th] lighting condition (Lambertian)

Pixel intensity at position x,y in i[th] image.

surface radiance $L(\theta_e,\phi_e)=\frac{1}{\pi} E\cos(\theta_s)\approx\cos(\theta_s) = \hat{n}\cdot\hat{s}$

$$I_i(x,y) = kL_i(x,y)$$
$$= k\rho(x,y)\hat{N}(x,y)\cdot\hat{S}_i$$

surface albedo

surface normal

i[th] light source direction

Combining all the measurements

$$\begin{pmatrix} I_1(x,y) \\ I_2(x,y) \\ \vdots \\ I_n(x,y) \end{pmatrix} = \begin{pmatrix} \hat{S}_1^T \\ \hat{S}_2^T \\ \vdots \\ \hat{S}_n^T \end{pmatrix} \rho(x,y)\hat{N}(x,y)$$

Solve for g(x,y). May be ill-conditioned

$$\begin{pmatrix} I_1(x,y) \\ I_2(x,y) \\ \vdots \\ I_n(x,y) \end{pmatrix} = \begin{pmatrix} \hat{S}_1^T \\ \hat{S}_2^T \\ \vdots \\ \hat{S}_n^T \end{pmatrix} \vec{g}(x,y)$$

A fix to avoid problems in dark areas: pre-multiply both sides by the image intensities

$$\begin{pmatrix} I_1(x,y) & 0 & \cdots & 0 \\ 0 & I_2(x,y) & \cdots & 0 \\ \vdots & \vdots & \vdots & 0 \\ 0 & 0 & 0 & I_n(x,y) \end{pmatrix} \begin{pmatrix} I_1(x,y) \\ I_2(x,y) \\ \vdots \\ I_n(x,y) \end{pmatrix}$$
$$= \begin{pmatrix} I_1(x,y) & 0 & \cdots & 0 \\ 0 & I_2(x,y) & \cdots & 0 \\ \vdots & \vdots & \vdots & 0 \\ 0 & 0 & 0 & I_n(x,y) \end{pmatrix} \begin{pmatrix} \hat{S}_1^T \\ \hat{S}_2^T \\ \vdots \\ \hat{S}_n^T \end{pmatrix} \vec{g}(x,y)$$

Recovering albedo and surface normal

$$\rho(x, y) = \left|\vec{g}(x, y)\right|$$

$$\hat{N}(x, y) = \frac{\vec{g}(x, y)}{\left|\vec{g}(x, y)\right|}$$

Surface shape from surface gradients

- Can you do it?
- What are the ambiguities? - Additive height constant
- What are the constraints?
  - no surface discontinuity – the surface is smooth
- So for your homework, we'll leave the computation at the gradients.