A13 – Photometric Stereo

Introduction

Surfaces are bright or dark due to:

1. their albedo or reflectance
2. the amount of light they are receiving.

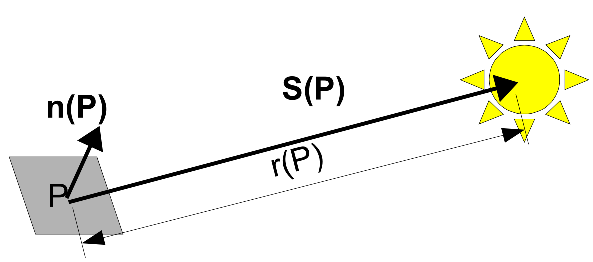
In this lesson we discuss the different sources and shadow models and apply them in extracting shape from shadow. Reflectance or albedo at a point P on a surface $\rho(P)$ is the ratio of the incident light over the reflected light.

Sources and Shading Models

An object is a light source if light coming from it is internally generated. Therefore, reflecting surfaces such as the moon, do not qualify in this definition. However, since they do affect the brightness of other surfaces they are known as secondary sources.

Point Source

Brightness from a point source drops by $\frac{1}{r^2}$. Let B(P) be brightness at point P $\hat{n}(P)$ normal vector at point P, $\vec{S}(P)$ a vector from P to the source and $r(P)$ the distance from P to the source.



Nearby Point Source

Example : Lightbulb in a room

For a nearby point source the $1/r^2$ dependence is pronounced. The brightness of a surface patch B(P) at point P is then

$$B(P)=\frac{\rho(P)\hat{n}(P)\cdot\vec{S}(P)}{r(P)^2} \quad . \tag{1}$$

Point Source at Infinity

Example : Sunlight

When the point source is at infinity, radial waves travelling from the source will appear as plane waves to the surface. Thus $\vec{S}(P)$ is constant everywhere, $\vec{S}_0$, and the $1/r^2$ dependence disappears. Thus

$$B(P)=\rho(P)\hat{n}(P)\cdot\vec{S}_0. \tag{2}$$

Line source

Example: Fluorescent Lamp, Xenon lamps in scanners. Dependence on distance drops off as $\frac{1}{r}$.

Area source

Example: X-ray view boxes, the sky on a cloudy day.

The amount of light a surface receives is integrated over the area of the source, considering each infinitesimal element of area as a point source.

Photometric Stereo

Consider a point source at infinity, source vector is $\vec{S}_1$.

Assume that intensity I captured by camera at point (x,y) is directly proportional to brightness of surface at that point, i.e.,

$$I(x,y)=kB(x,y) \tag{3}$$

then

$$\begin{aligned} I(x,y) &= k\,\rho(x,y)\,\hat{n}(x,y)\cdot\vec{S}_1 \\ &= \vec{g}(x,y)\cdot\vec{V}_1 \end{aligned} \tag{4}$$

with

$$\vec{g}(x,y) = \rho(x,y)\,\hat{n}(x,y) \tag{5}$$

$$\vec{V}_1 = k\,\vec{S}_1 \quad . \tag{6}$$

We can estimate the shape of the surface by capturing multiple images of the surface with the sources at different locations. The information about the surface will be coded in the shadings obtained from the images.

Let there be N sources in 3-d space. We can define a matrix

$$\mathbf{V} = \begin{bmatrix} V_{11} & V_{12} & V_{13} \\ V_{21} & V_{22} & V_{23} \\ \ldots & \ldots & \ldots \\ V_{N1} & V_{N2} & V_{N3} \end{bmatrix} \tag{7}$$

where each row is a source, each column is the x,y,z component of the source. If we take N images of the surface using each of these N sources, then for each point (x,y) on the surface we have

$$\begin{aligned} I_1(x,y) &= V_{11}g_1 + V_{12}g_2 + V_{13}g_3 \\ I_2(x,y) &= V_{21}g_1 + V_{22}g_2 + V_{23}g_3 \\ I_N(x,y) &= V_{N1}g_1 + V_{N2}g_2 + V_{N3}g_3 \end{aligned} \tag{8}$$

Or in matrix form

$$\mathbf{I} = \mathbf{V}\,\mathbf{g} \quad . \tag{9}$$

Since I and V are known, we can solve for g in the least squares sense by

$$\mathbf{g} = (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T\mathbf{I} \tag{10}$$

To get the normal vector we simplify normalize g by its length

$$\hat{n} = \frac{\mathbf{g}}{|\mathbf{g}|} \quad . \tag{11}$$

To get the shape from normals, note that if the surface elevation is expressed as

$$z = f(x,y) \tag{12}$$

we rewrite this as

c(x,y,z) = z – f(x,y) = 0 and get the gradient to get the normal vector

$$\nabla c(x,y,z)=\frac{-\partial f}{\partial x}\hat{\mathbf{i}}-\frac{\partial f}{\partial y}\hat{\mathbf{j}}+\hat{\mathbf{k}} \quad . \tag{13}$$

Once the surface normals (nx, ny, nz) are estimated using photometric stereo, they are related to the partial derivative of f as

$$\frac{\partial f}{\partial x}=\frac{-nx}{nz};\frac{\partial f}{\partial y}=\frac{-ny}{nz} \tag{14}$$

The surface elevation z at point (u,v) is given by f(u,v) and is evaluated by a line integral

$$f(u,v)=\int_0^u\left(\frac{\partial f}{\partial x}\right)dx+\int_u^v\left(\frac{\partial f}{\partial y}\right)dy \tag{15}$$

Procedure

1. Load the matlab file `photos.mat` which contains 4 images I1, I2, I3, and I4. The images are synthetic spherical surfaces illuminated by a far away point source located respectively at

   V1 = {0.085832, 0.17365, 0.98106}

   V2 = {0.085832, -0.17365, 0.98106}

   V3 = {0.17365, 0, 0.98481}

   V4 = {0.16318, -0.34202, 0.92542}

   Use `loadmatfile` in Scilab.

2. Compute the surface normals using Equation 10 and 11.
3. From the surface normals compute the elevation z=f(u,v) and display a 3D plot of the object shape. There are many ways to compute the line integral to get the 3d surface. You may try the Frankot-Chellappa algorithm [1], the Fast Marching Method [2], or outright line integration [3]. Plot the 3D shape recovered as a mesh.

Reference

[1] Frankot, R. T., & Chellappa, R. (1988). A method for enforcing integrability in shape from

shading algorithms. IEEE Transactions on pattern analysis and machine intelligence, 10(4), 439-451.

[2] Ho, J., Lim, J., Yang, M. H., & Kriegman, D. (2006, May). Integrating surface normal vectors using fast marching method. In European Conference on Computer Vision (pp. 239-250). Springer, Berlin, Heidelberg.

[3]Wu, Z., & Li, L. (1988). A line-integration based method for depth recovery from surface normals. Computer Vision, Graphics, and Image Processing, 43(1), 53-66.

