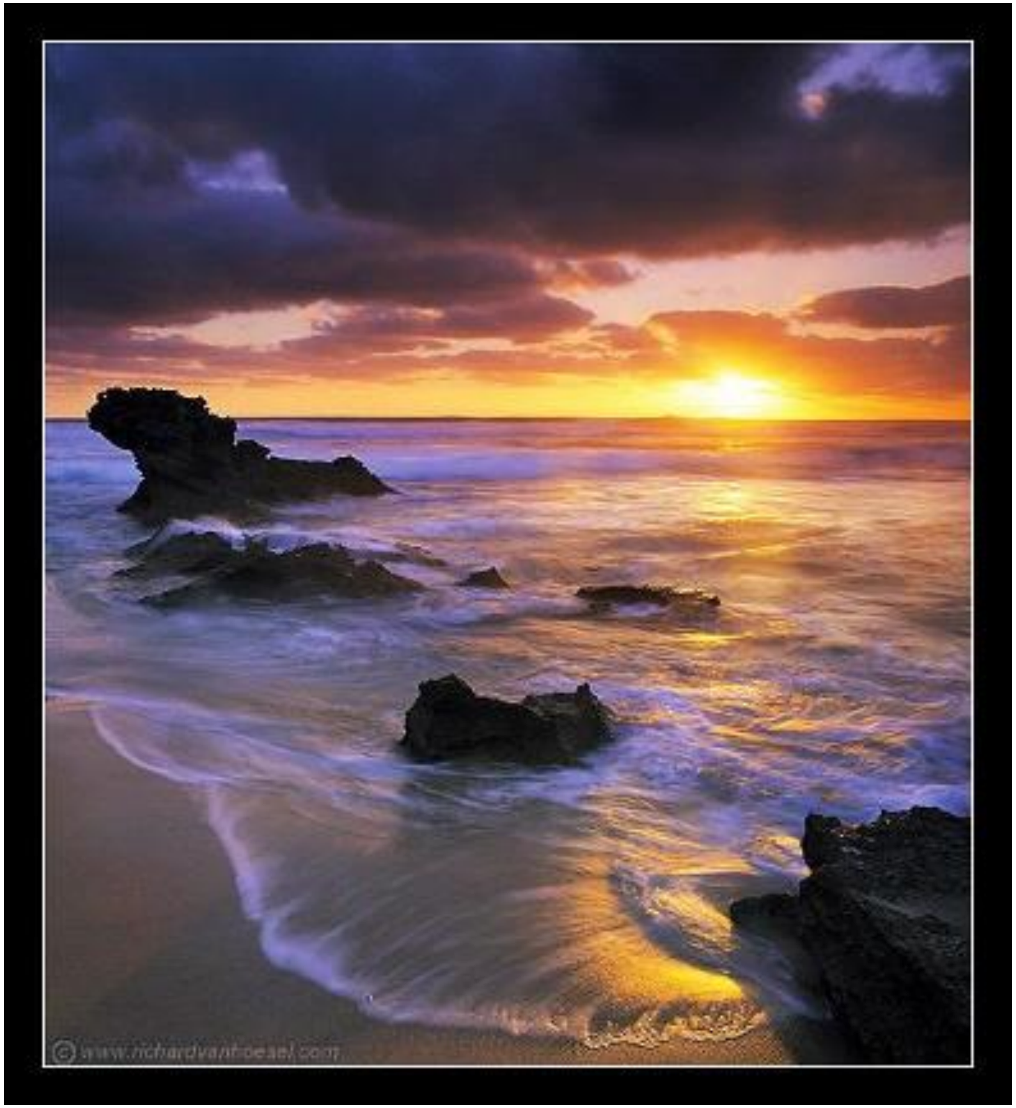
Light and shading



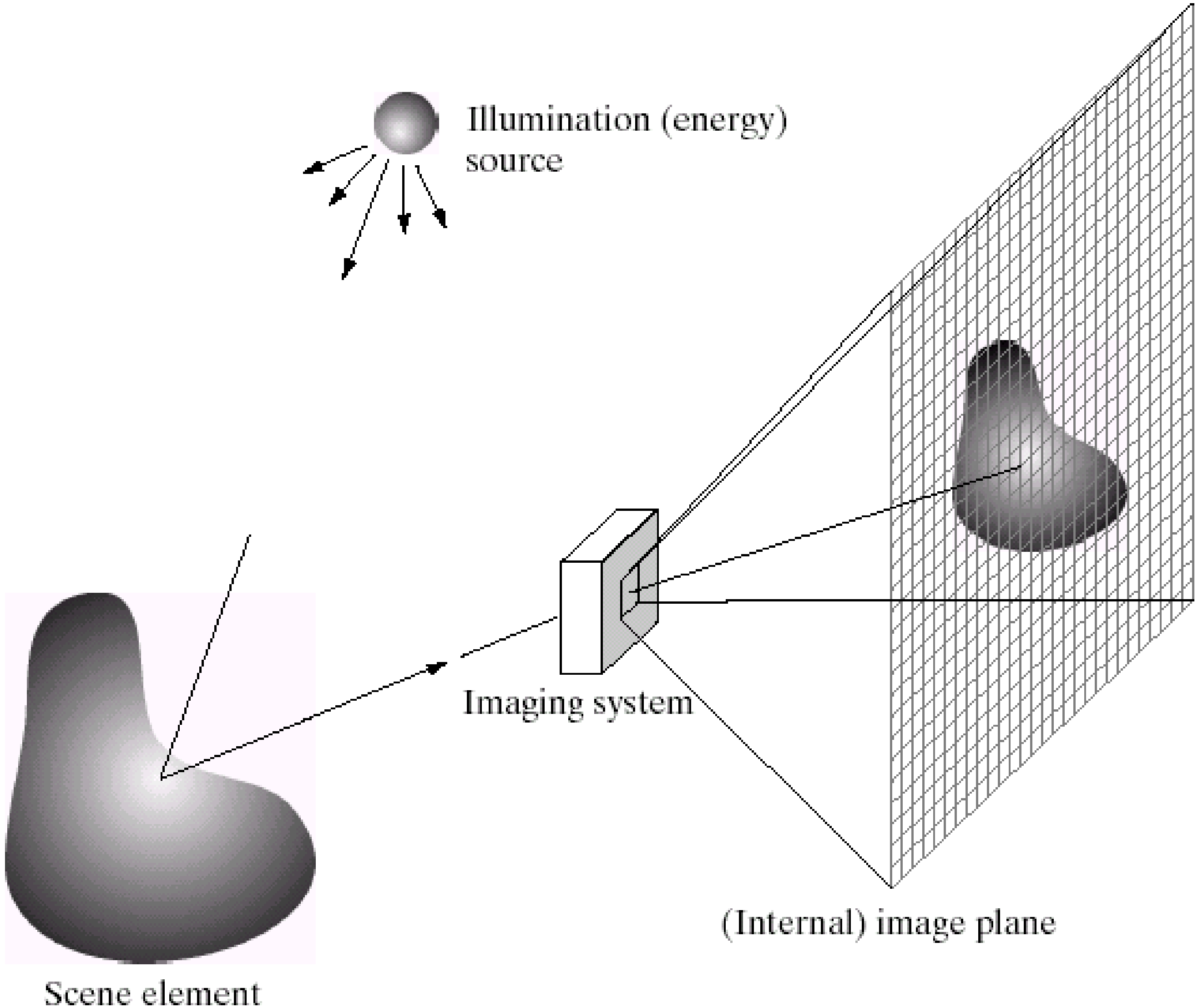
What determines a pixel's intensity?

What can we infer about the scene from pixel intensities?

Image Source: A. Efros

Some slides credited to James Hays@Brown University, Kristen Grauman@UT Austin and Svatalana Lazebnik@UIUC

How light is recorded



Digital camera

A digital camera replaces film with a sensor array

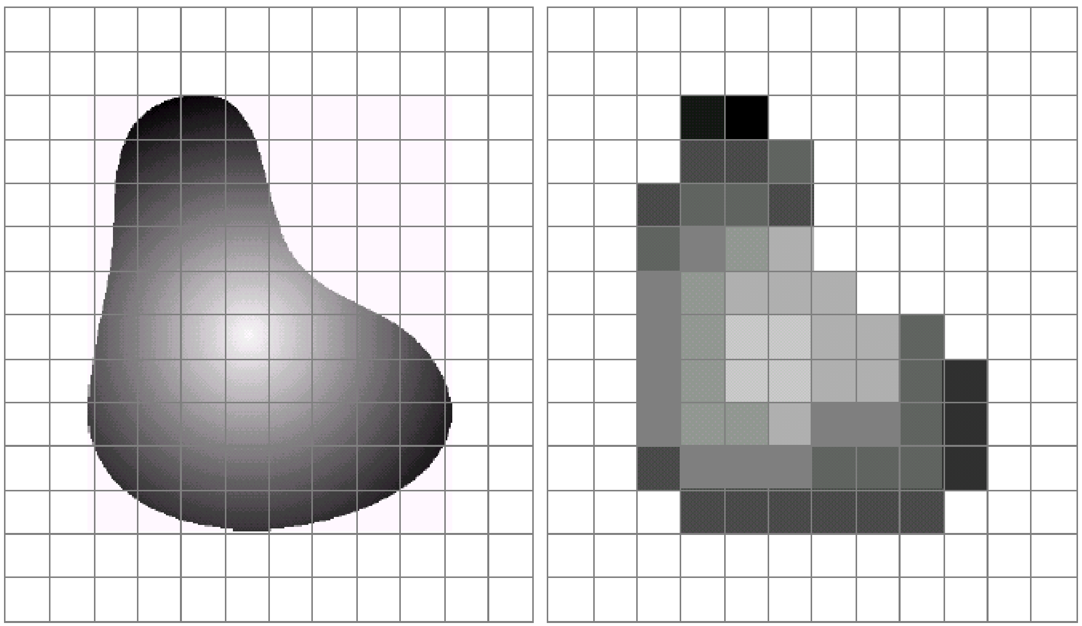
Each cell in the array is light-sensitive diode that converts photons to electrons

Two common types: Charge Coupled Device (CCD) and CMOS

http://electronics.howstuffworks.com/digital-camera.htm

Slide by Steve Seitz

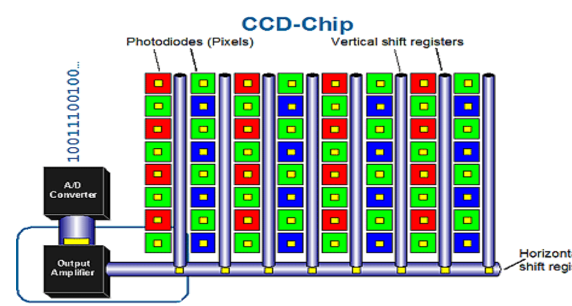
Sensor Array



a b

FIGURE 2.17 (a) Continuos image projected onto a sensor array. (b) Result of image sampling and quantization.

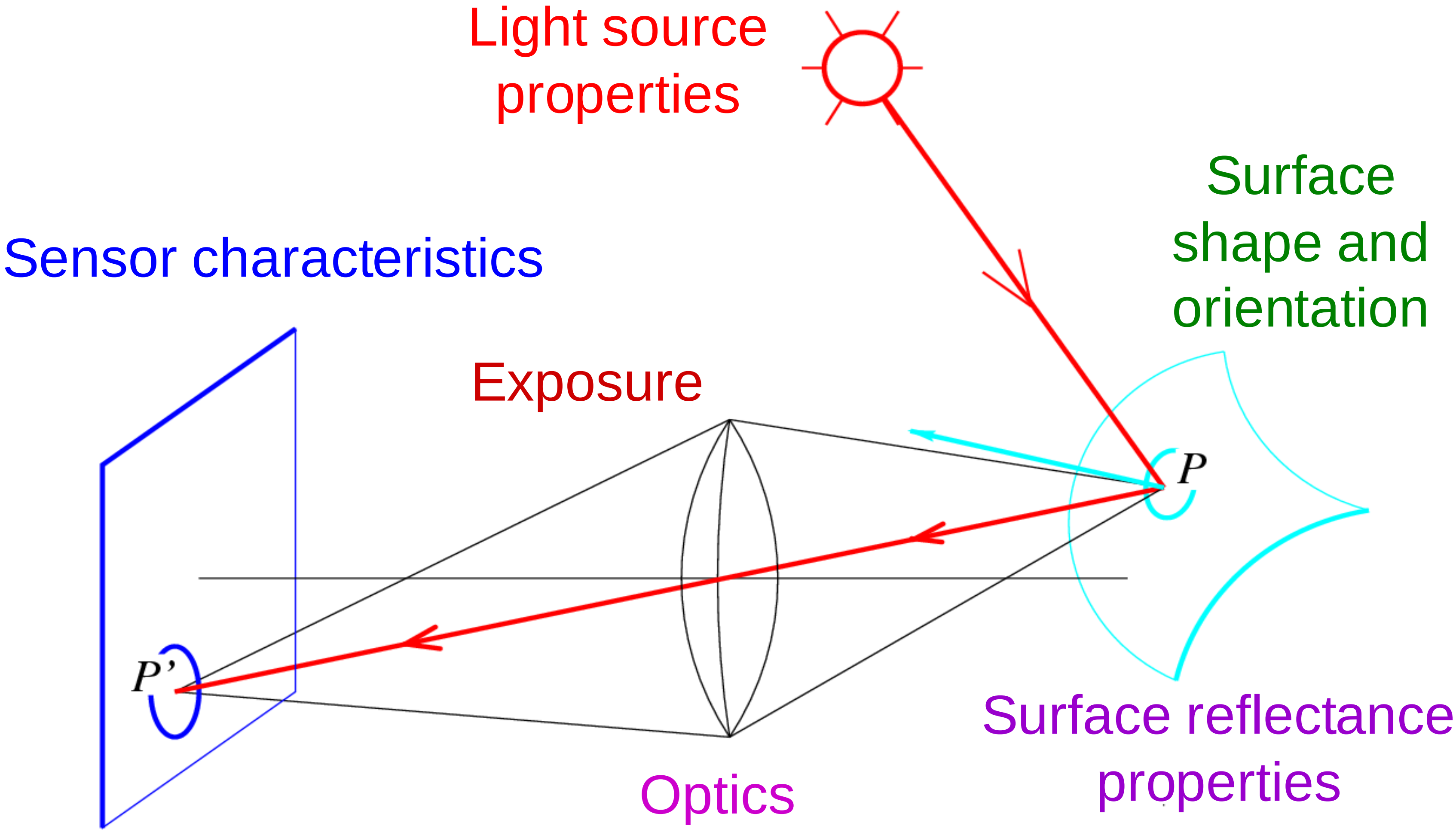
CMOS sensor



Each sensor cell record a small range of orientations
amount of light coming in

Image formation

What determines the brightness of an image pixel?



Slide by L. Fei-Fei

Intensity and Surface Orientation

Intensity depends on illumination angle because less light comes in at oblique angles.

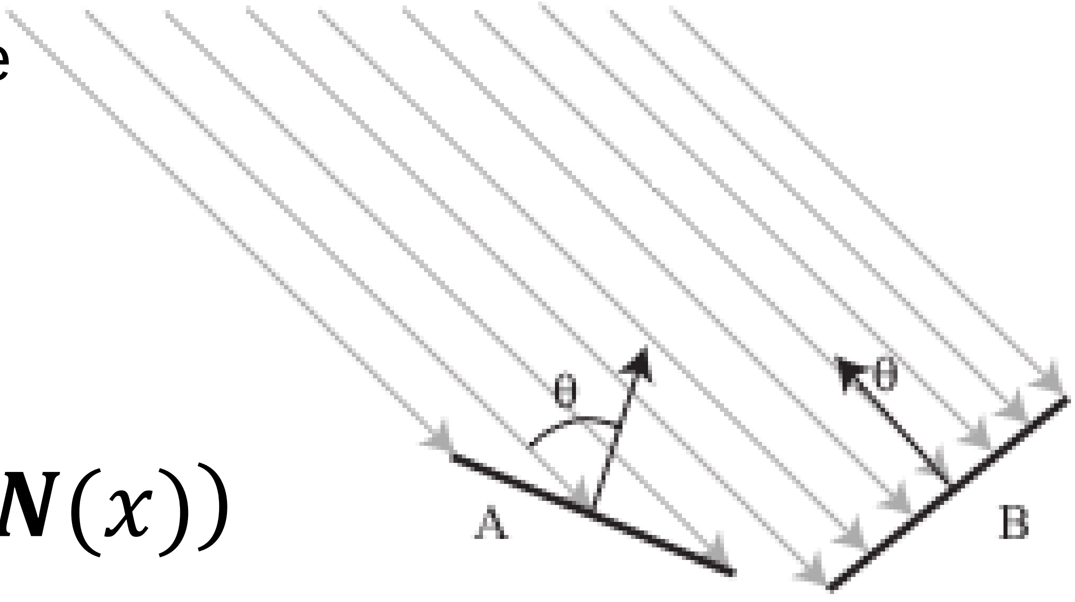
ρ = albedo

$\boldsymbol{S}$ = directional source

$\boldsymbol{N}$ = surface normal

I = reflected intensity

$$I(x) = \rho(x)(\boldsymbol{S} \cdot \boldsymbol{N}(x))$$



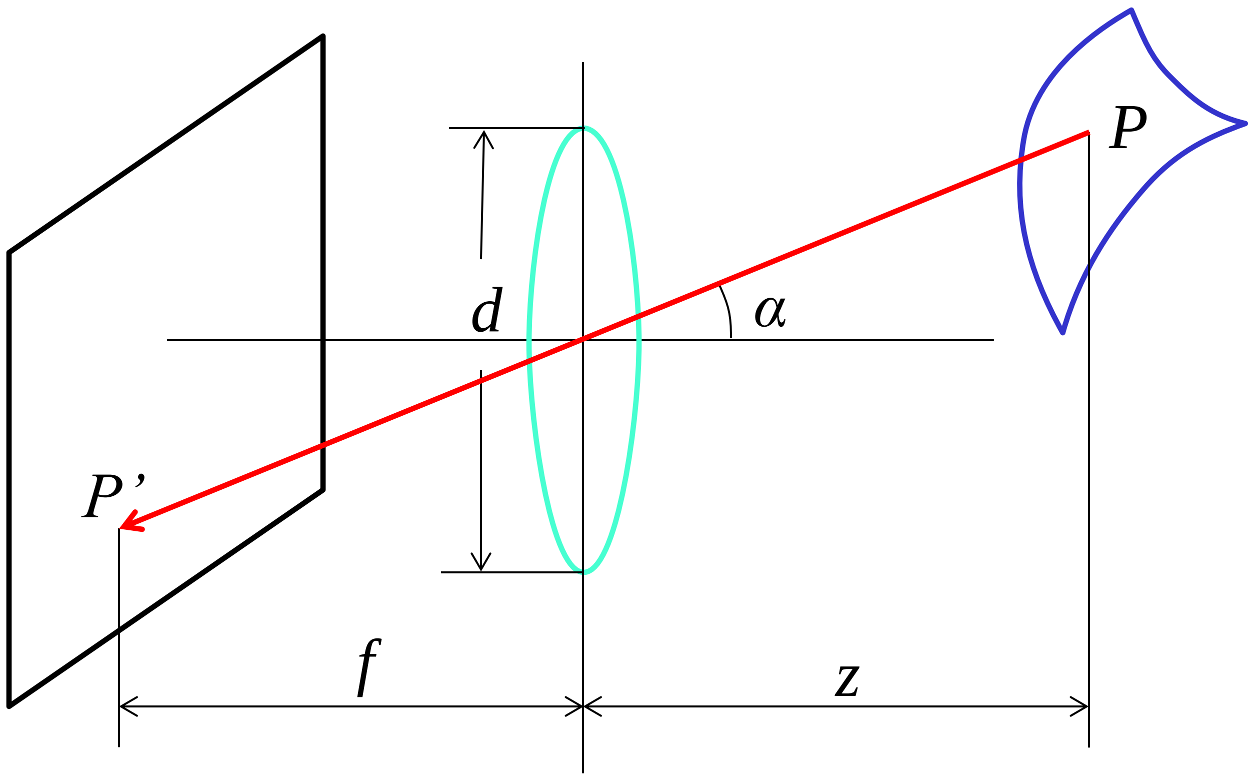
Fundamental radiometric relation

L: *Radiance* emitted from *P* toward *P*’ (辐射度)

- Energy carried by a ray (Watts per sq. meter per steradian)

E: *Irradiance* falling on *P*’ from the lens (辐照度)

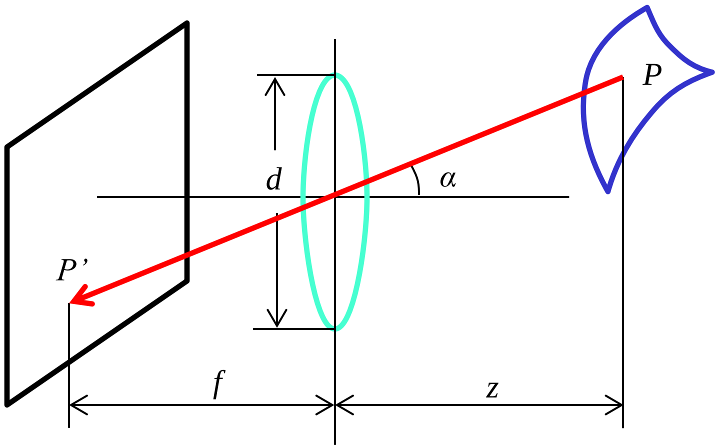
- Energy arriving at a surface (Watts per sq. meter)



What is the relationship between *E* and *L*?

Szeliski 2.2.3

Fundamental radiometric relation

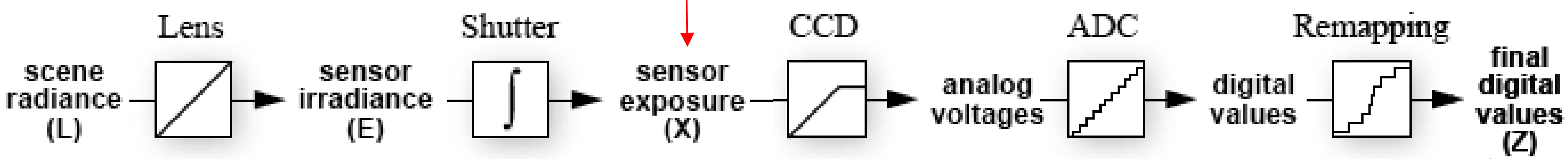


$$E = \left[\frac{\pi}{4} \left(\frac{d}{f} \right)^2 \cos^4 \alpha \right] L$$

- Image irradiance is linearly related to scene radiance
- Irradiance is proportional to the area of the lens and inversely proportional to the squared distance between the lens and the image plane
- The irradiance falls off as the angle between the viewing ray and the optical axis increases (natural vignetting)

From light rays to pixel values

$$X = E \cdot \Delta t$$



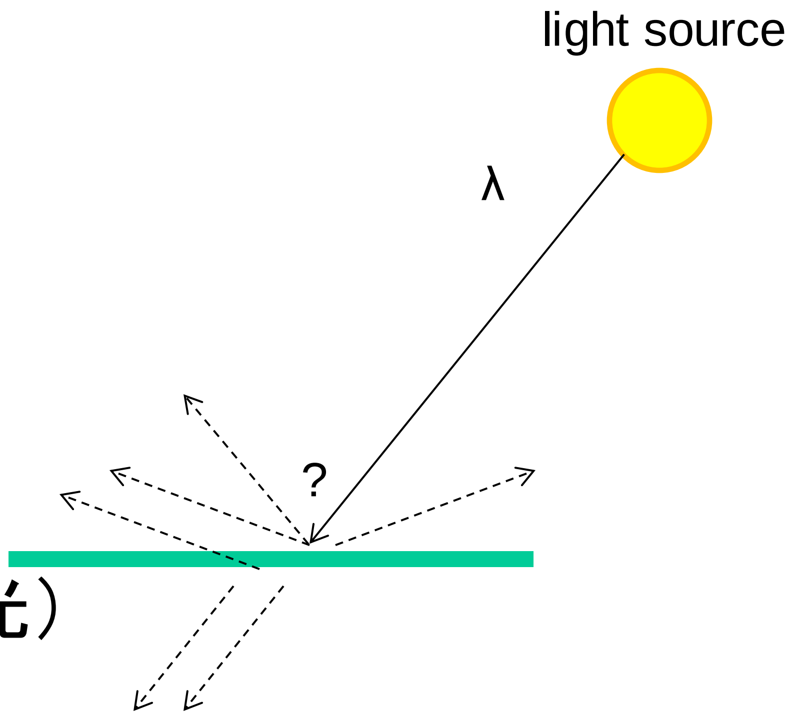
$$E = \left[\frac{\pi}{4} \left(\frac{d}{f} \right)^2 \cos^4 \alpha \right] L$$

$$Z = f\left(E \cdot \Delta t\right)$$

- Camera response function: the mapping f from irradiance to pixel values
  - Useful if we want to estimate material properties
  - Enables us to create high dynamic range images
  - For more info: P. E. Debevec and J. Malik, [Recovering High Dynamic Range Radiance Maps from Photographs](), *SIGGRAPH 97*

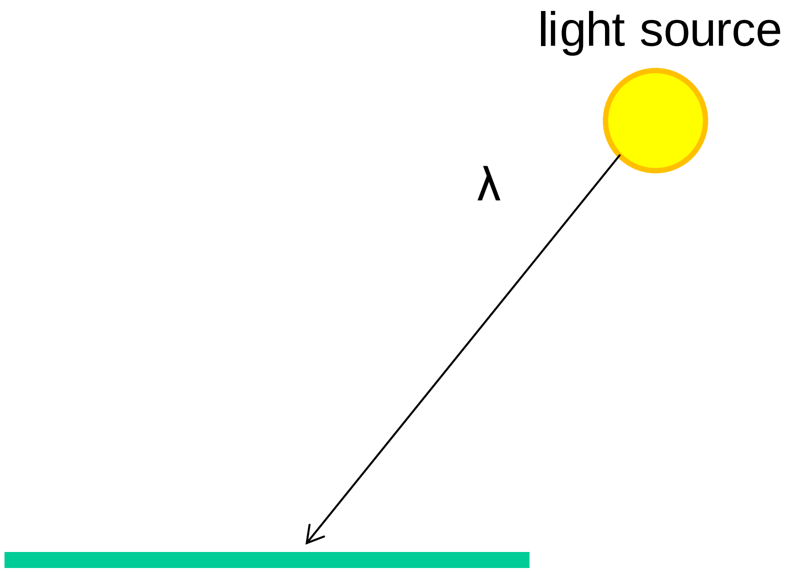
A photon's life choices

- Absorption
- Diffusion
- Reflection
- Transparency
- Refraction
- Fluorescence （荧光 ）
- Subsurface scattering
- Phosphorescence （磷光）
- Interreflection



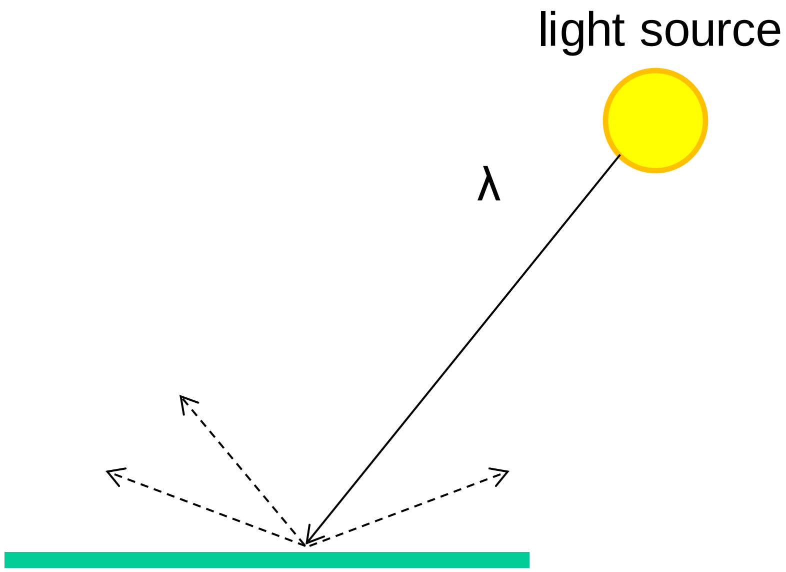
A photon's life choices

- **Absorption**
- Diffusion
- Reflection
- Transparency
- Refraction
- Fluorescence
- Subsurface scattering
- Phosphorescence
- Interreflection



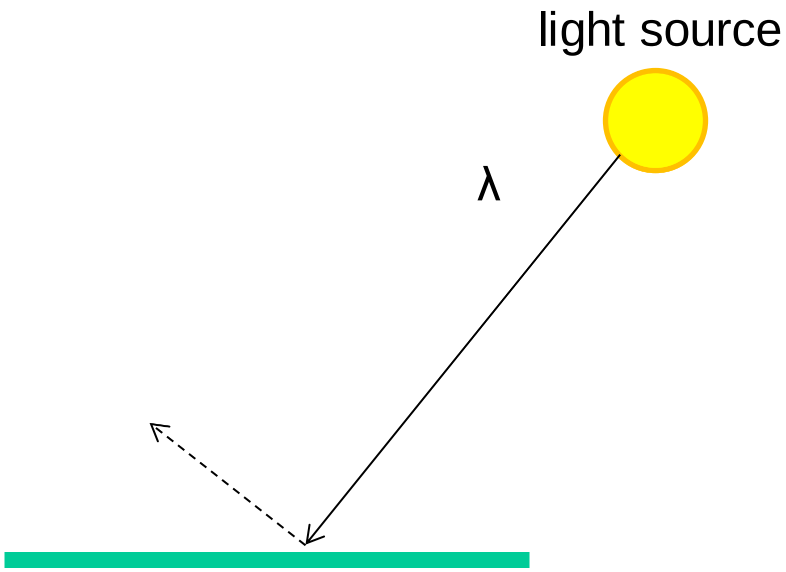
A photon's life choices

- Absorption
- **Diffuse Reflection**
- Reflection
- Transparency
- Refraction
- Fluorescence
- Subsurface scattering
- Phosphorescence
- Interreflection



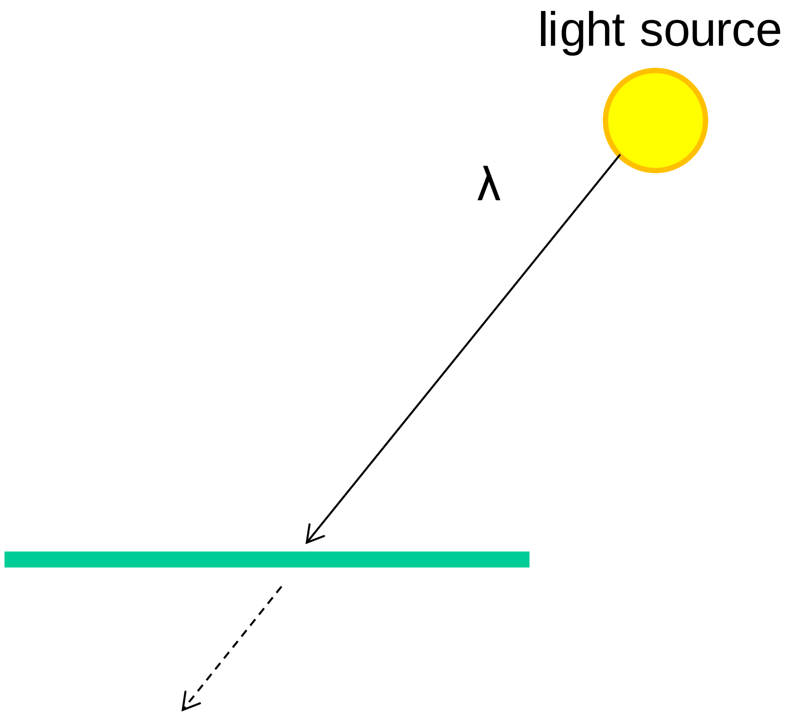
A photon's life choices

- Absorption
- Diffusion
- **Specular Reflection**
- Transparency
- Refraction
- Fluorescence
- Subsurface scattering
- Phosphorescence
- Interreflection



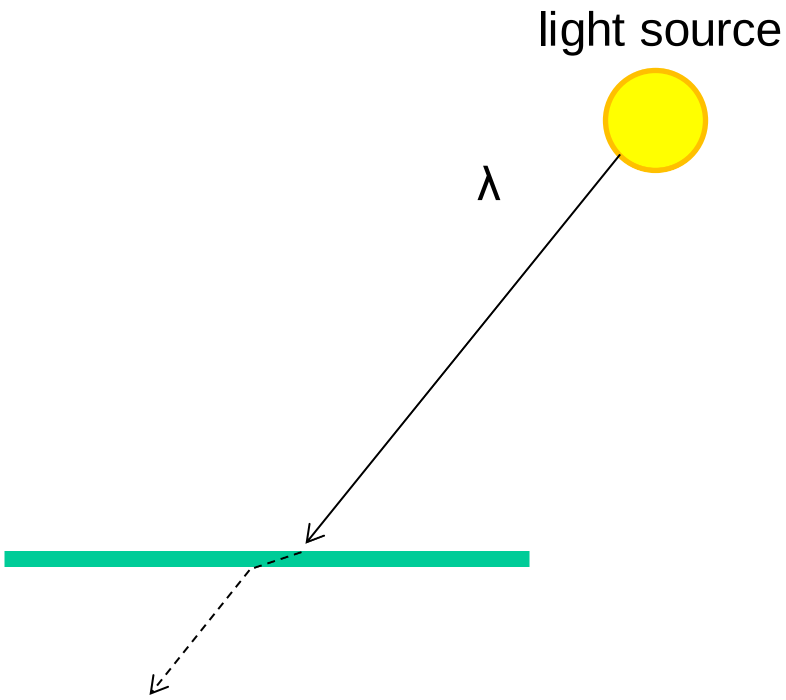
A photon's life choices

- Absorption
- Diffusion
- Reflection
- **Transparency**
- Refraction
- Fluorescence
- Subsurface scattering
- Phosphorescence
- Interreflection



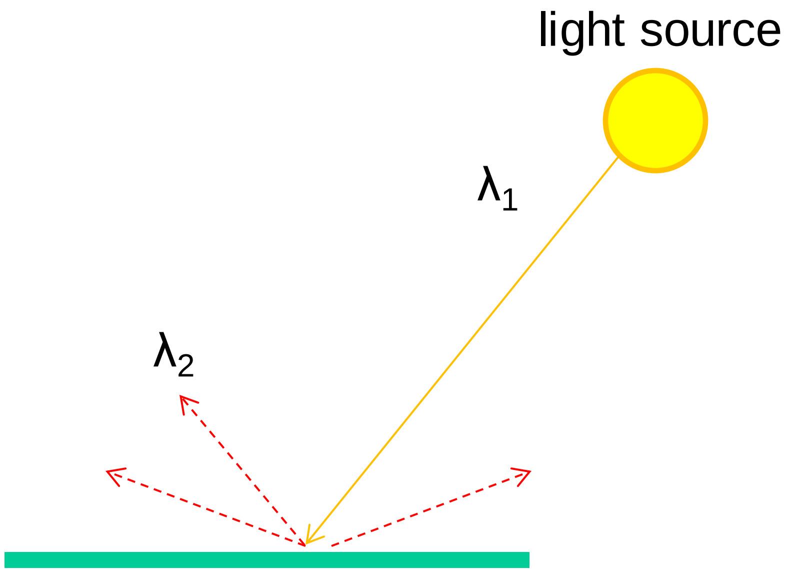
A photon's life choices

- Absorption
- Diffusion
- Reflection
- Transparency
- **Refraction**
- Fluorescence
- Subsurface scattering
- Phosphorescence
- Interreflection



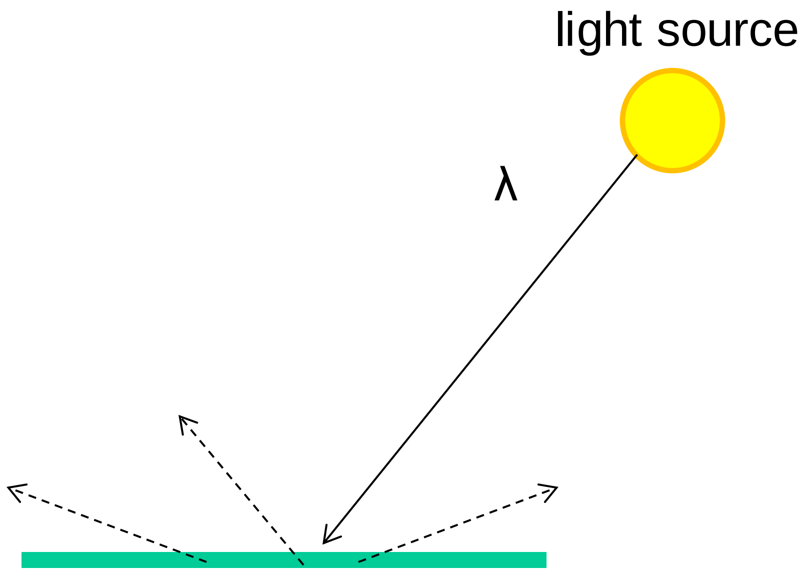
A photon's life choices

- Absorption
- Diffusion
- Reflection
- Transparency
- Refraction
- **Fluorescence**
- Subsurface scattering
- Phosphorescence
- Interreflection



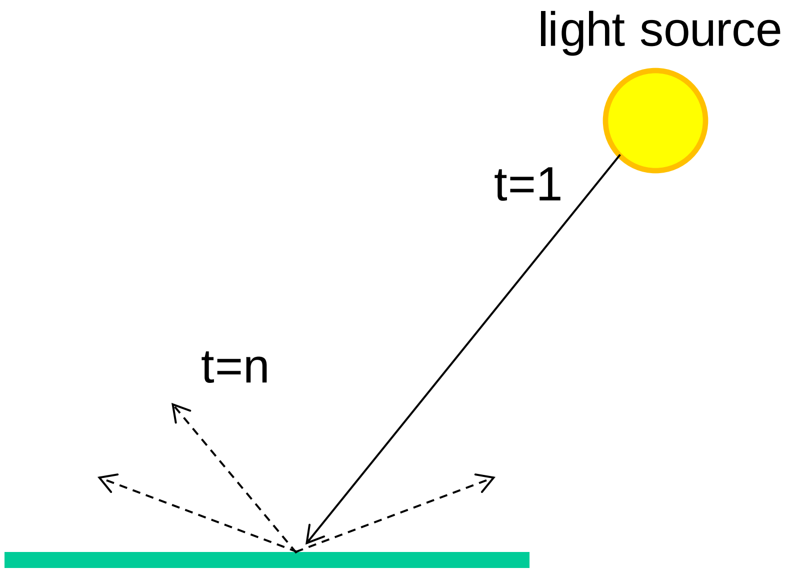
A photon's life choices

- Absorption
- Diffusion
- Reflection
- Transparency
- Refraction
- Fluorescence
- **Subsurface scattering**
- Phosphorescence
- Interreflection



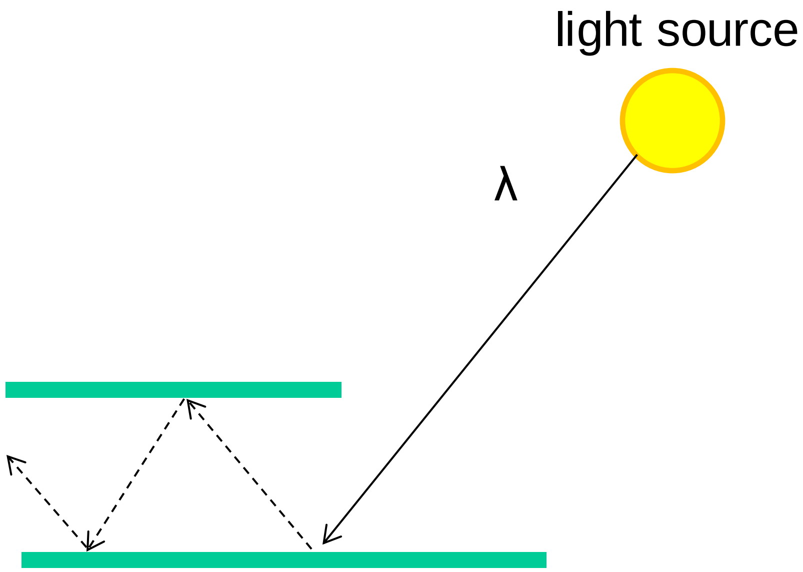
A photon's life choices

- Absorption
- Diffusion
- Reflection
- Transparency
- Refraction
- Fluorescence
- Subsurface scattering
- **Phosphorescence**
- Interreflection



A photon's life choices

- Absorption
- Diffusion
- Reflection
- Transparency
- Refraction
- Fluorescence
- Subsurface scattering
- Phosphorescence
- **Interreflection**

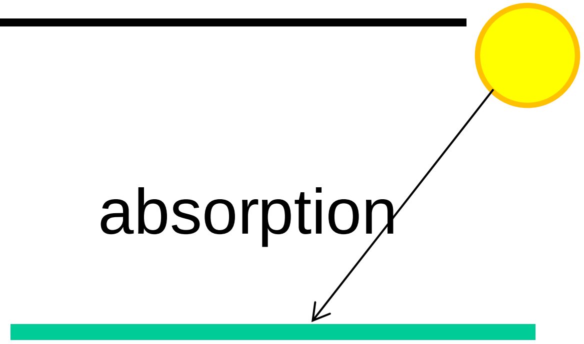


(Specular Interreflection)

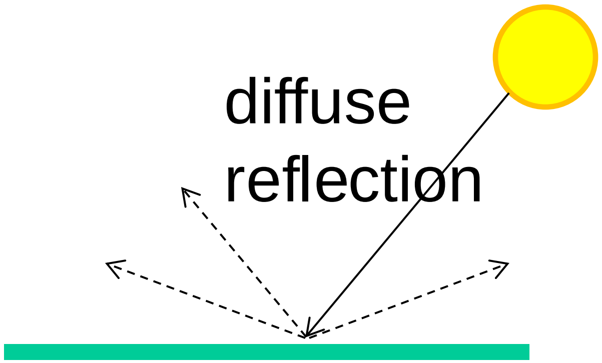
Some common effects

When light hits a typical surface

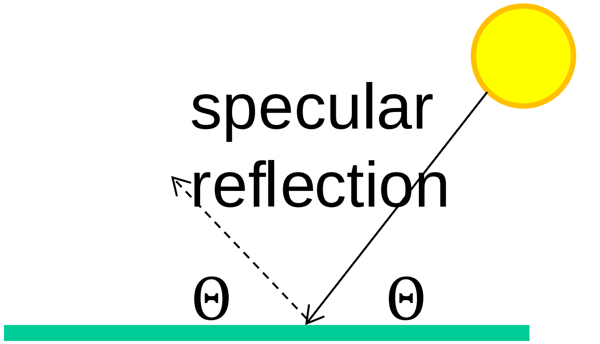
- Some light is absorbed (1-ρ)
  - More absorbed for low albedos



- Some light is reflected diffusely
  - Independent of viewing direction

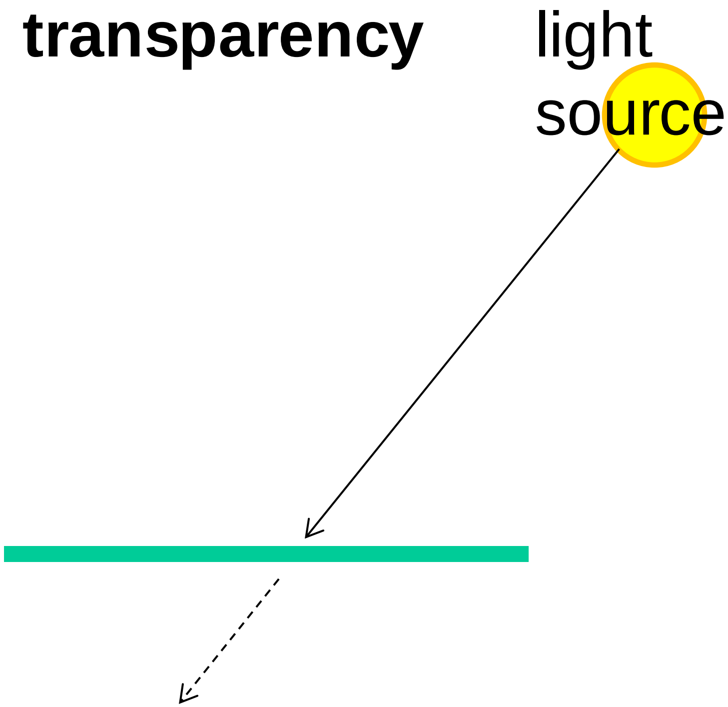


- Some light is reflected specularly
  - Light bounces off (like a mirror), depends on viewing direction

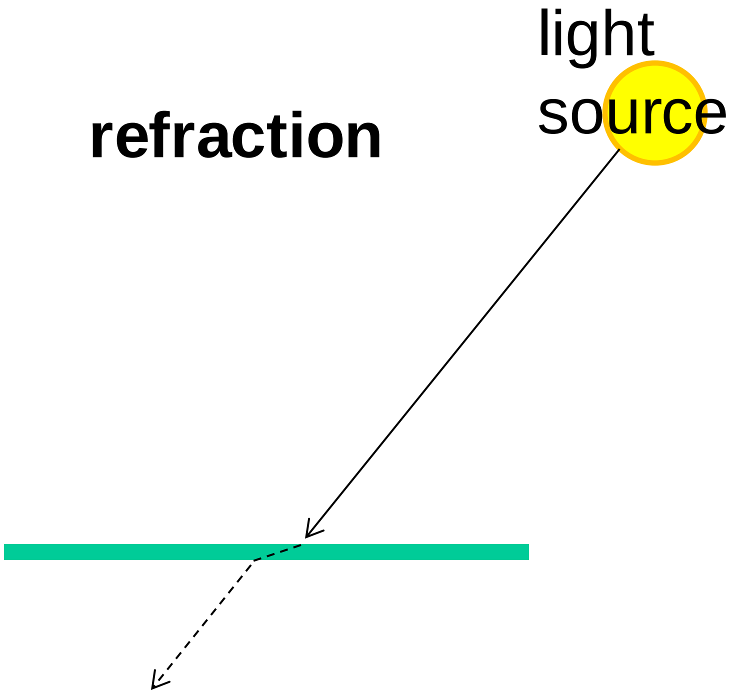


Other possible effects

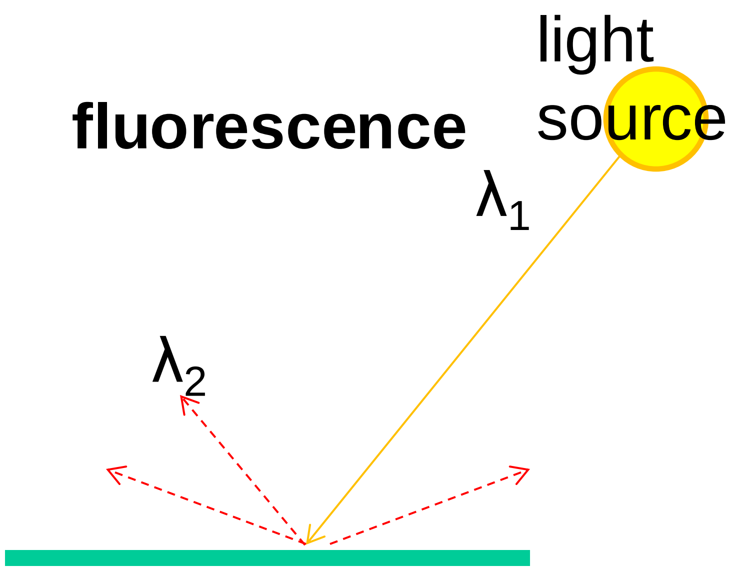
transparency



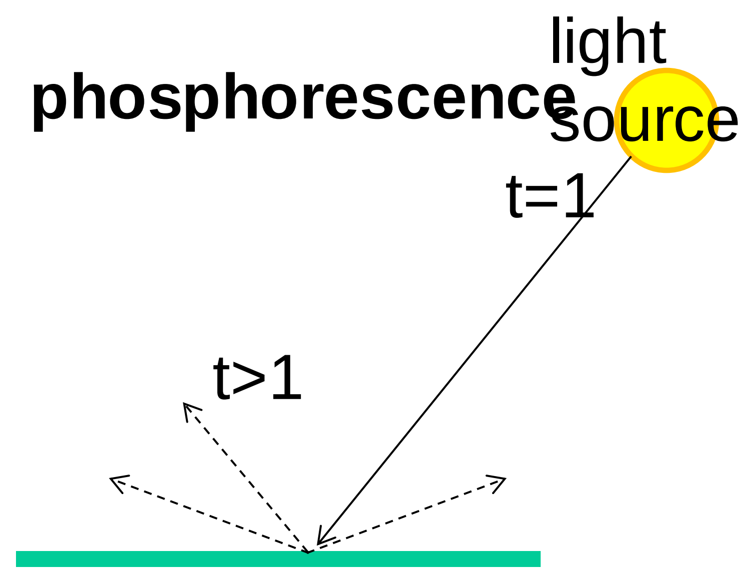
refraction



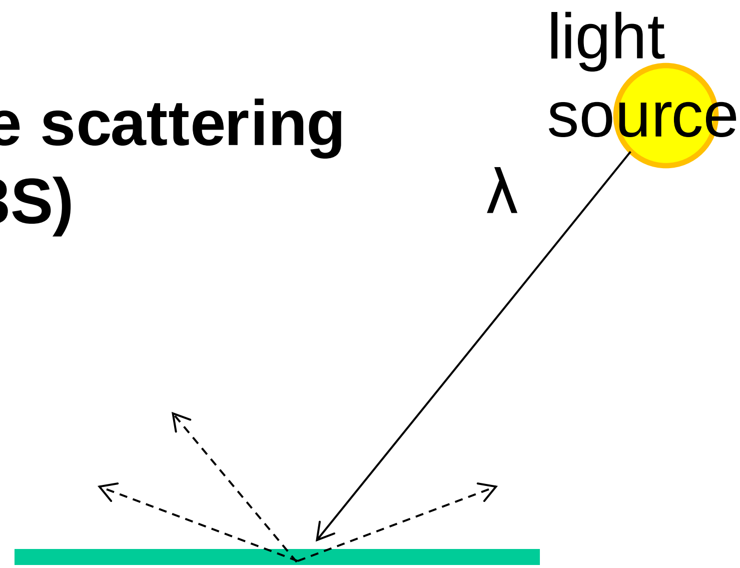
fluorescence



phosphorescence

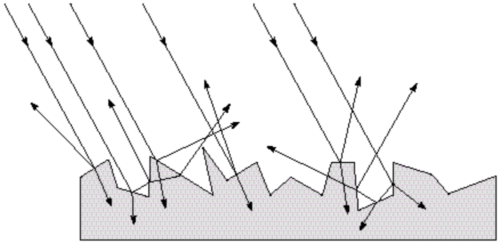


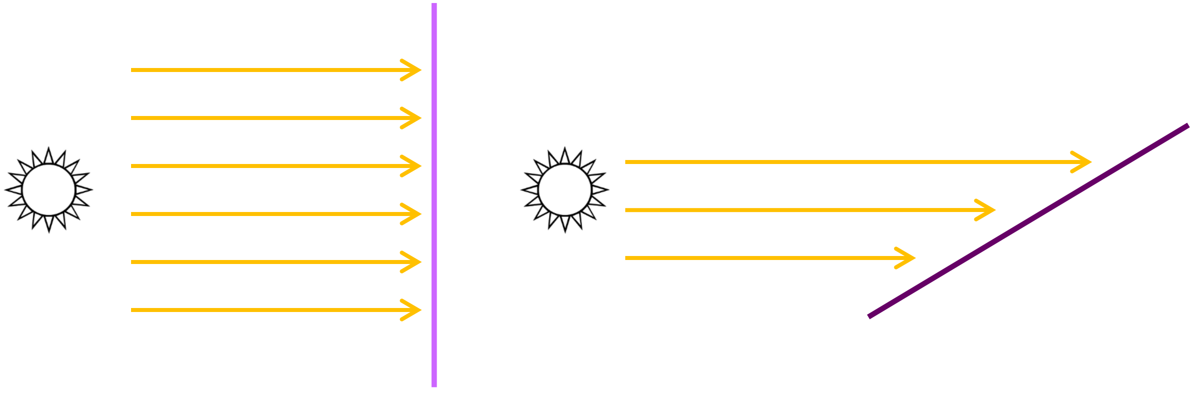
subsurface scattering (3S)



Diffuse reflection

- Light is reflected equally in all directions
  - Dull, matte surfaces like chalk or latex paint
  - Microfacets scatter incoming light randomly
  - Effect is that light is reflected equally in all directions
- Brightness of the surface depends on the incidence of illumination





Photometric stereo (shape from shading)

- Can we reconstruct the shape of an object based on shading cues?

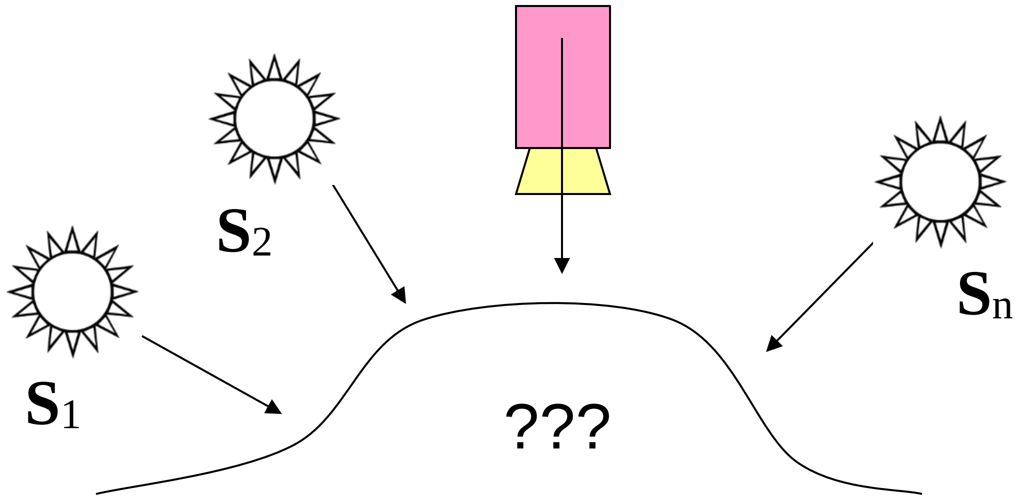
Luca della Robbia, *Cantoria*, 1438

Photometric stereo

Assume:

- A Lambertian object
- A *local shading model* (each point on a surface receives light only from sources visible at that point)
- A set of *known* light source directions
- A set of pictures of an object, obtained in exactly the same camera/object configuration but using different sources

Goal: reconstruct object shape and albedo



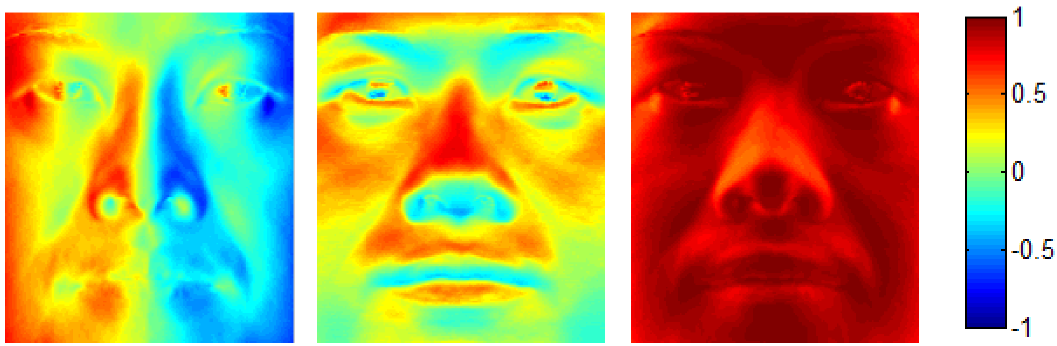
F&P 2nd ed., sec. 2.2.4

Example

Input

Recovered albedo

Recovered normal field



x y z

Recovered surface model

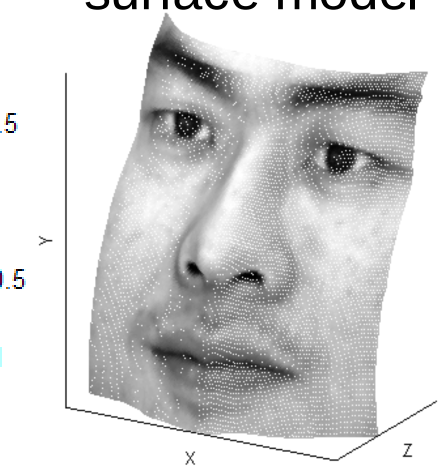
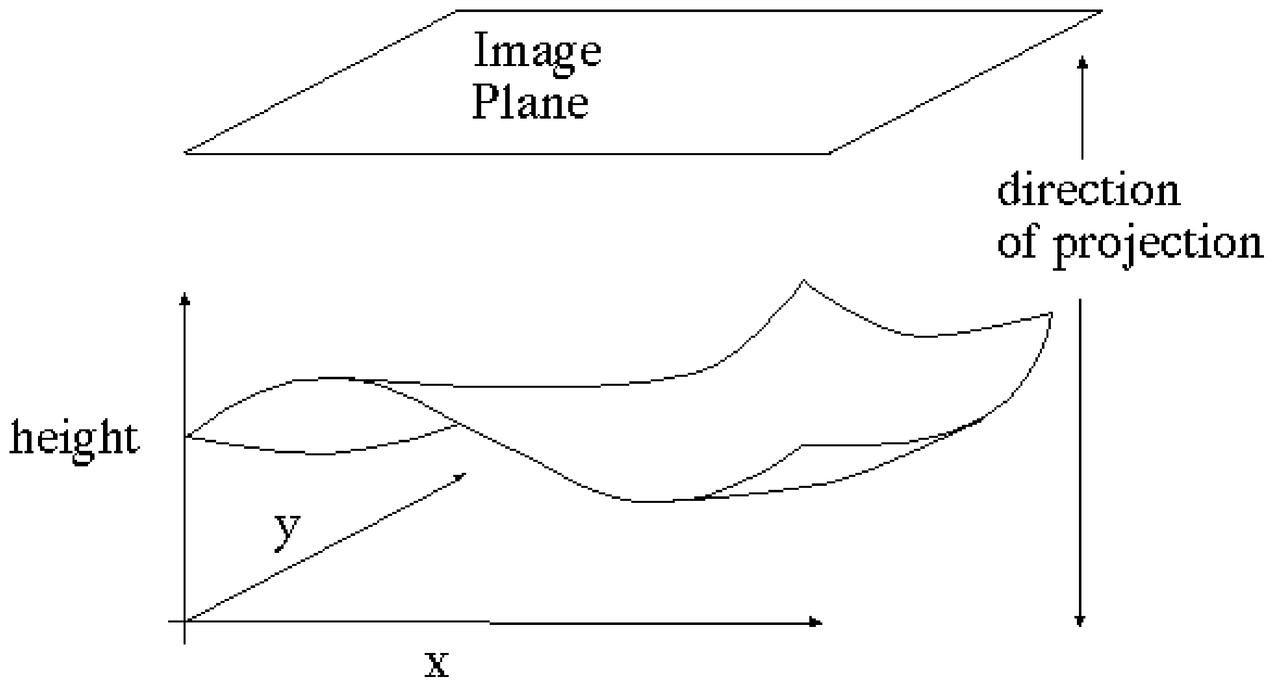


Image model

- **Known:** source vectors $\mathbf{S}_j$ and pixel values $I_j(x,y)$
- **Unknown:** surface normal $\mathbf{N}(x,y)$ and albedo 反照率 $\rho(x,y)$



F&P 2nd ed., sec. 2.2.4

Image model

- **Known:** source vectors $\mathbf{S}_j$ and pixel values $I_j(x,y)$
- **Unknown:** surface normal $\mathbf{N}(x,y)$ and albedo 反照率 $\rho(x,y)$
- Assume that the response function of the camera is a linear scaling by a factor of k
- Lambert's law:

$$
\begin{aligned}
I_j(x,y) &= k\rho(x,y)\big(\mathbf{N}(x,y)\cdot\mathbf{S}_j\big) \\
&= \big(\rho(x,y)\mathbf{N}(x,y)\big)\cdot(k\mathbf{S}_j) \\
&= \mathbf{g}(x,y)\cdot\mathbf{V}_j
\end{aligned}
$$

F&P 2[nd] ed., sec. 2.2.4

Least squares problem

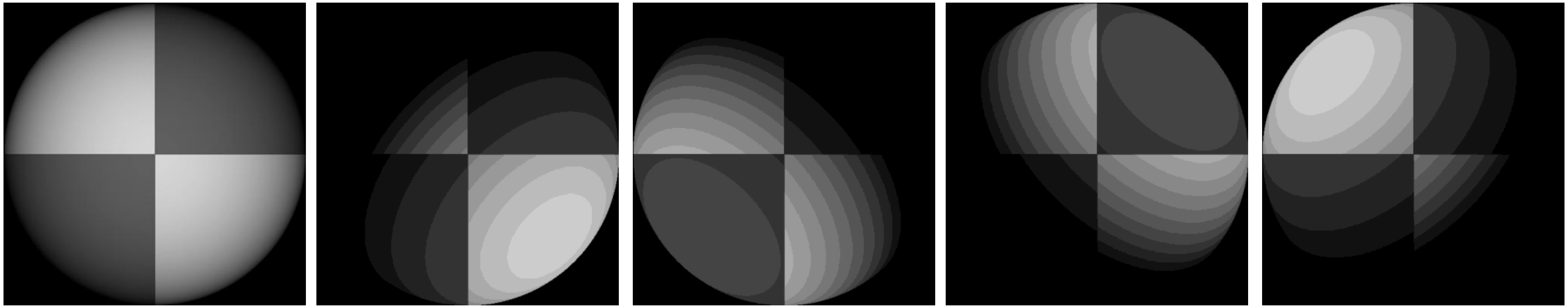
- For each pixel, set up a linear system:

$$\underbrace{\begin{bmatrix} I_1(x,y) \\ I_2(x,y) \\ \vdots \\ I_n(x,y) \end{bmatrix}}_{\substack{(n \times 1) \\ \text{known}}} = \underbrace{\begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \\ \vdots \\ \mathbf{V}_n^T \end{bmatrix}}_{\substack{(n \times 3) \\ \text{known}}} \underbrace{\mathbf{g}(x,y)}_{\substack{(3 \times 1) \\ \text{unknown}}}$$

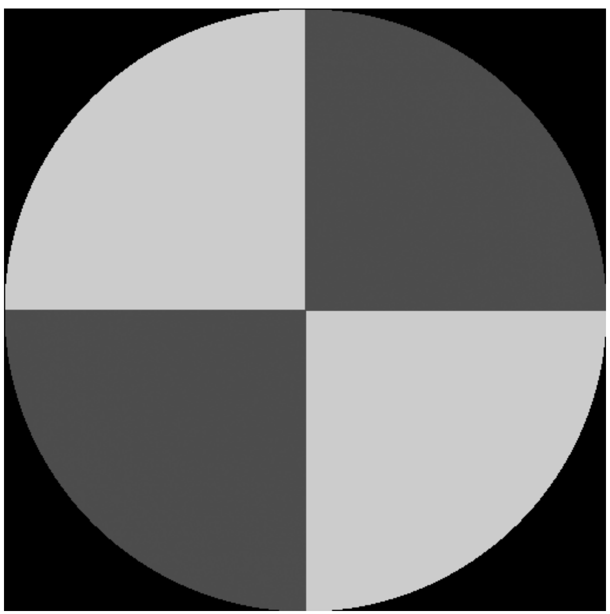
- Obtain least-squares solution for $\mathbf{g}(x,y)$ (which we defined as $\mathbf{N}(x,y)\ \rho(x,y)$)
- Since $\mathbf{N}(x,y)$ is the unit normal, $\rho(x,y)$ is given by the magnitude of $\mathbf{g}(x,y)$
- Finally, $\mathbf{N}(x,y) = \mathbf{g}(x,y) / \rho(x,y)$

F&P 2nd ed., sec. 2.2.4

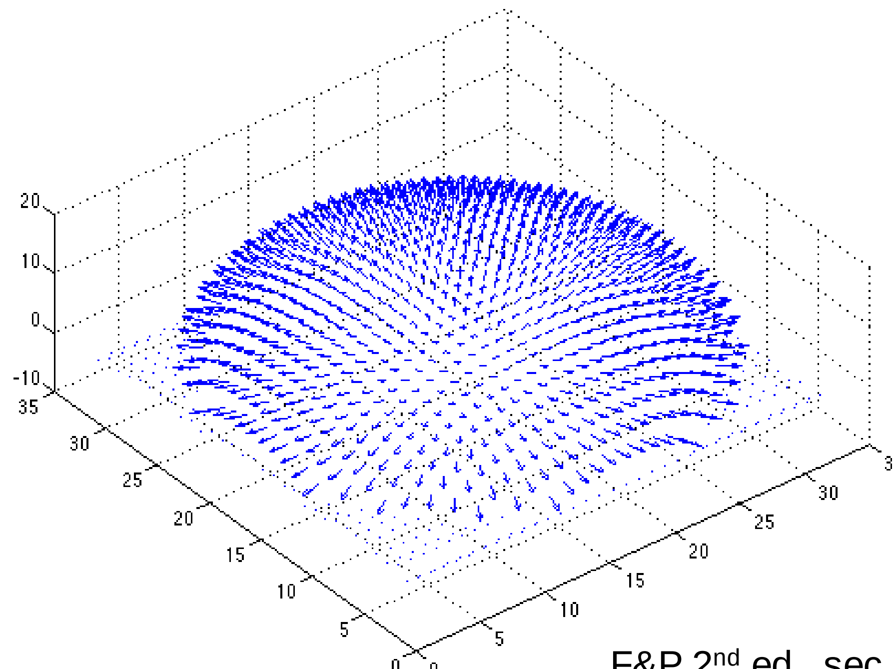
Synthetic example



Recovered albedo



Recovered normal field



F&P 2nd ed., sec. 2.2.4

Recovering a surface from normals

Given a normal vector field

$$\mathbf{N} = (N_x, N_y, N_z)$$

we wish to recover the surface z(x, y)

The gradients of (z) can be derived from N

$$[\frac{\partial z}{\partial x} = \frac{N_x}{N_z}, \quad \frac{\partial z}{\partial y} = \frac{N_y}{N_z}]$$

Integration of gradients can be performed directly using numerical

$$[z(x, y) = \int \left(\frac{\partial z}{\partial x} dx + \frac{\partial z}{\partial y} dy \right)]$$

Recovering a surface from normals

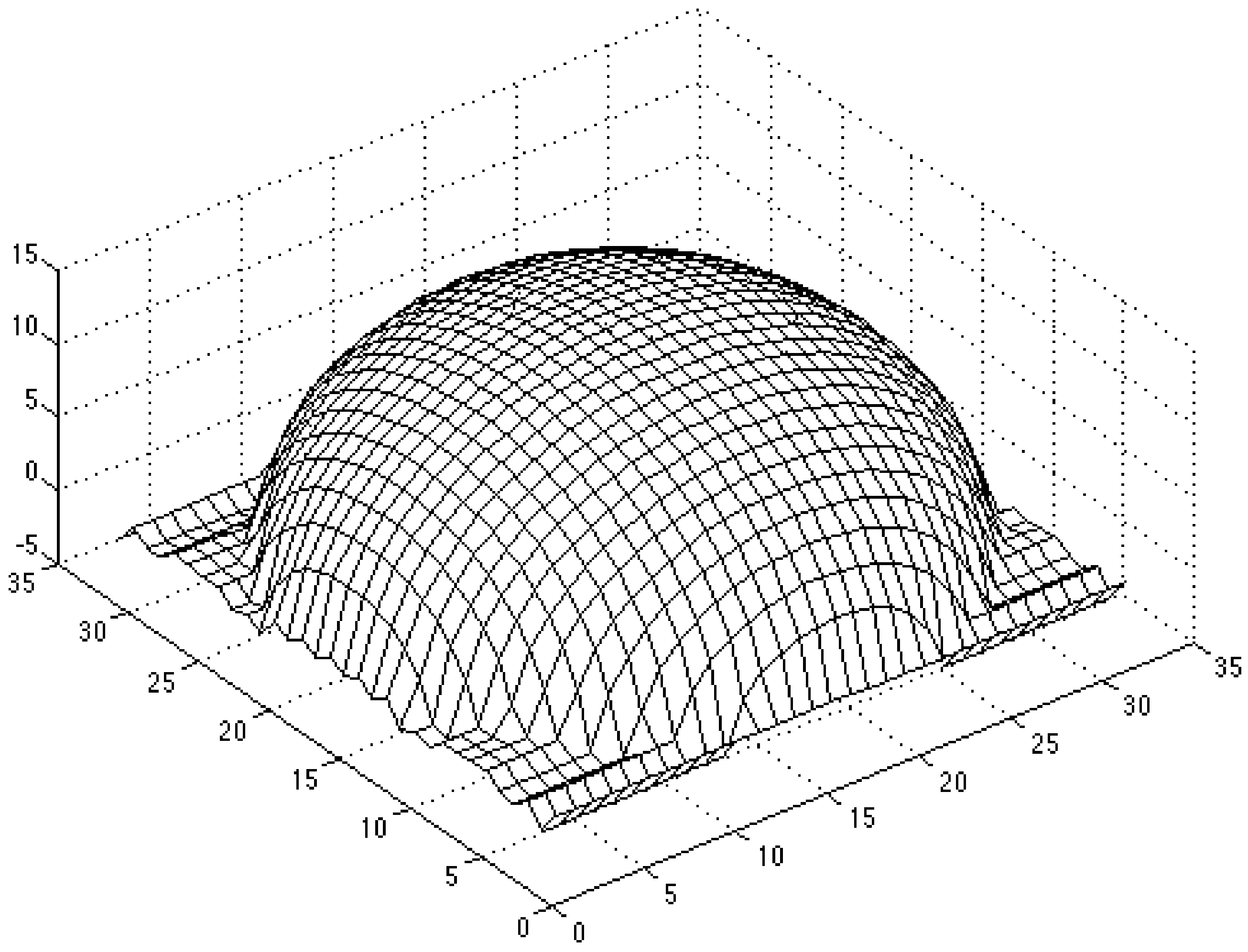
```
Input: Normal fields Nx, Ny, Nz
Output: Height map Z

# Normalize the normal vectors
Nx_normalized = Nx / Nz
Ny_normalized = Ny / Nz

# Initialize height map Z
Z = zeros_like(Nx)

# Perform numerical integration
for x in range(width):
    for y in range(height):
        if x > 0:
            Z[x, y] += Z[x-1, y] + Nx_normalized[x, y]
        if y > 0:
            Z[x, y] += Z[x, y-1] + Ny_normalized[x, y]
```

Surface recovered by integration



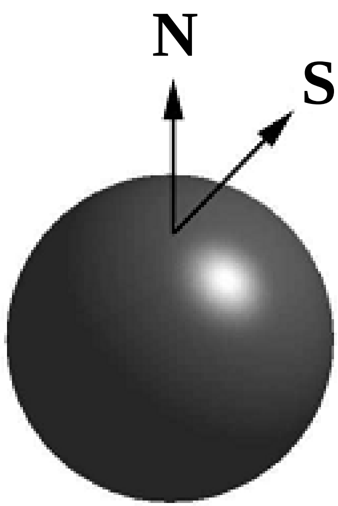
F&P 2nd ed., sec. 2.2.4

Limitations

- Simplistic reflectance and lighting model
- No shadows
- No interreflections
- No missing data
- Integration is tricky

Finding the direction of the light source

$$\boxed{I(x,y) = \mathbf{N}(x,y) \cdot \mathbf{S}(x,y)}$$



Full 3D case:

$$\begin{pmatrix} N_x(x_1,y_1) & N_y(x_1,y_1) & N_z(x_1,y_1) \\ N_x(x_2,y_2) & N_y(x_2,y_2) & N_z(x_2,y_2) \\ \vdots & \vdots & \vdots \\ N_x(x_n,y_n) & N_y(x_n,y_n) & N_z(x_n,y_n) \end{pmatrix} \begin{pmatrix} S_x \\ S_y \\ S_z \end{pmatrix} = \begin{pmatrix} I(x_1,y_1) \\ I(x_2,y_2) \\ \vdots \\ I(x_n,y_n) \end{pmatrix}$$

For points on the *occluding contour*:

$$\begin{pmatrix} N_x(x_1,y_1) & N_y(x_1,y_1) \\ N_x(x_2,y_2) & N_y(x_2,y_2) \\ \vdots & \vdots \\ N_x(x_n,y_n) & N_y(x_n,y_n) \end{pmatrix} \begin{pmatrix} S_x \\ S_y \end{pmatrix} = \begin{pmatrix} I(x_1,y_1) \\ I(x_2,y_2) \\ \vdots \\ I(x_n,y_n) \end{pmatrix}$$

P. Nillius and J.-O. Eklundh, “Automatic estimation of the projected light source direction,” CVPR 2001

Finding the direction of the light source

P. Nillius and J.-O. Eklundh, “Automatic estimation of the projected light source direction,” CVPR 2001

Application: Detecting composite photos

Fake photo

Real photo

M. K. Johnson and H. Farid, [Exposing Digital Forgeries by Detecting Inconsistencies in Lighting](), ACM Multimedia and Security Workshop, 2005.