Basic Ray-Tracing Ideas -II

D.A. Forsyth, UIUC

Recall: Randomized estimates of integrals

Weak law of large numbers

$$\text{if} \qquad x_i \sim p(x)$$
$$\text{then} \quad \frac{1}{N}\sum_{i=1}^{N} f(x_i) \rightarrow \int f(x)p(x)dx$$

- i.e. we can approximate integrals with sums
  - example: p(x) uniform, stochastic sampling of pixel
- generically, known as Monte Carlo estimates

Importance weighting

if $x_i \sim p(x)$

then $\frac{1}{N}\sum_{i=1}^{N} f(x_i) \to \int f(x)p(x)dx$

If $x_i \sim p(x)$

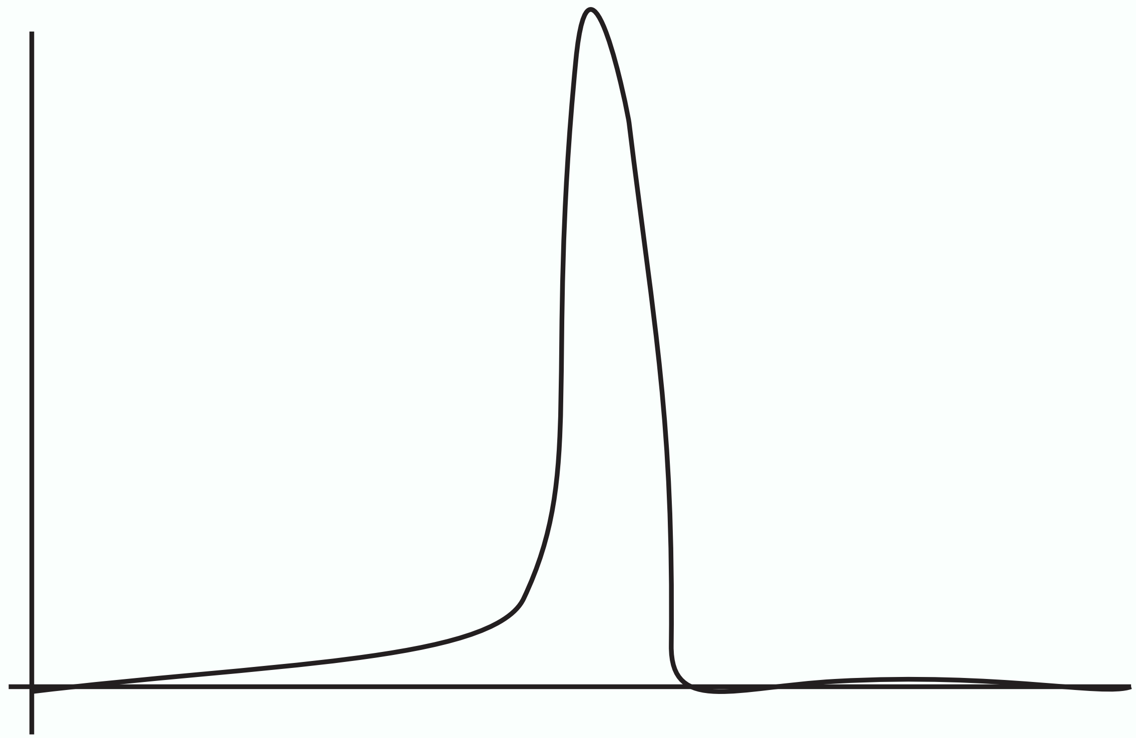
Then $$\int f(x)dx \approx \frac{1}{N}\sum_i \frac{f(x_i)}{p(x_i)}$$

Randomized estimates and variance

$$\int f(x)dx \approx \frac{1}{N}\sum_i \frac{f(x_i)}{p(x_i)}$$

- The estimate is the value of a random variable
  - (different random samples -> different estimates)
  - whose expected value is the value of the integral
  - but whose variance might be very big
    - and is usually very hard to know
- Simple reasoning suggests that
  - p should be big when f is big, etc.

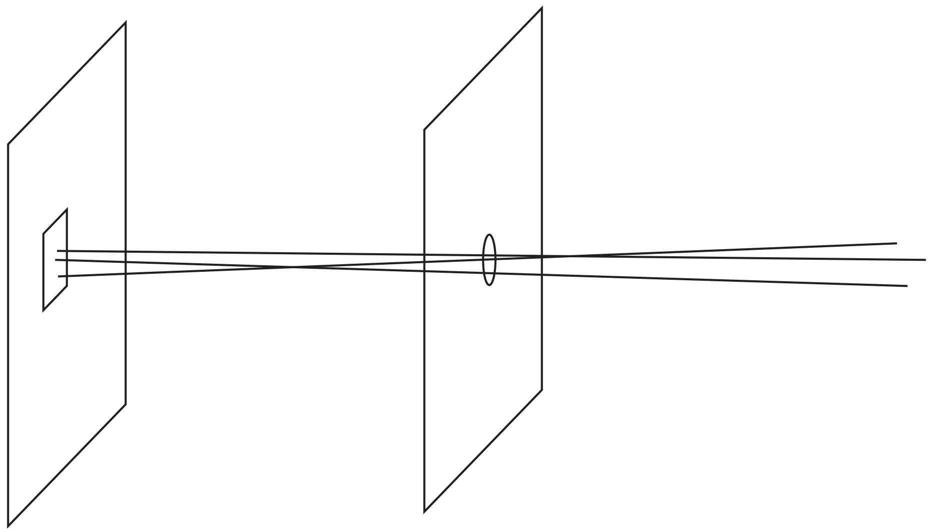
Variance example



Drawing uniform samples will get a poor estimate
-> draw samples mostly at peak and downweight

Algorithmic framework

$$v = \int_\Lambda \int_D \int_\Omega \int_T w(\mathbf{x}, \lambda, \omega, t) L(\mathbf{x}, \omega, t)\, dt\, d\omega\, dx\, d\lambda \approx \sum_{i \in \text{rays}} g(\text{ray}) L(\text{ray})$$

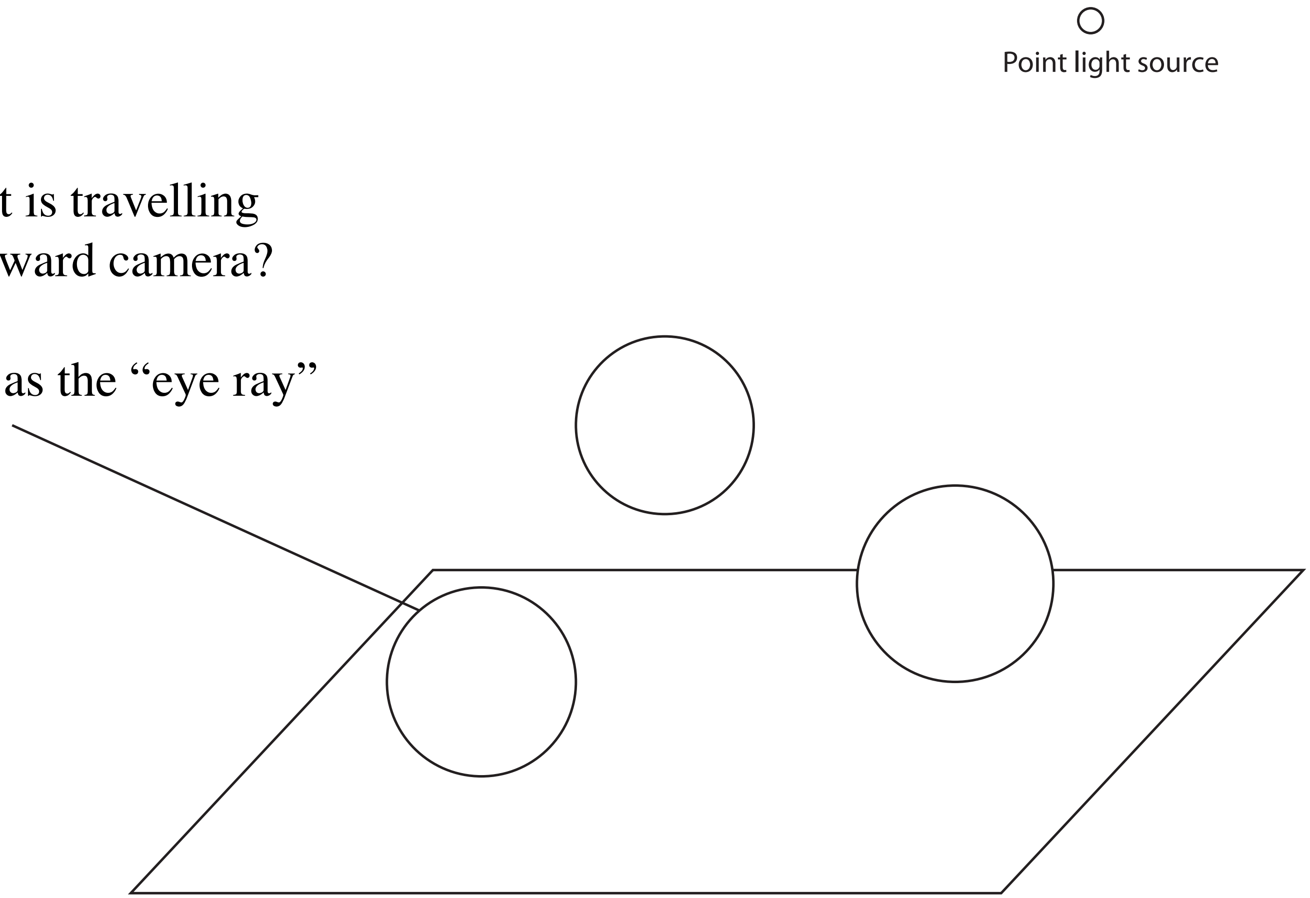


Computational problem - what is L(ray)?

Very simple ray-tracing

How much light is travelling
down this ray toward camera?

sometimes known as the “eye ray”

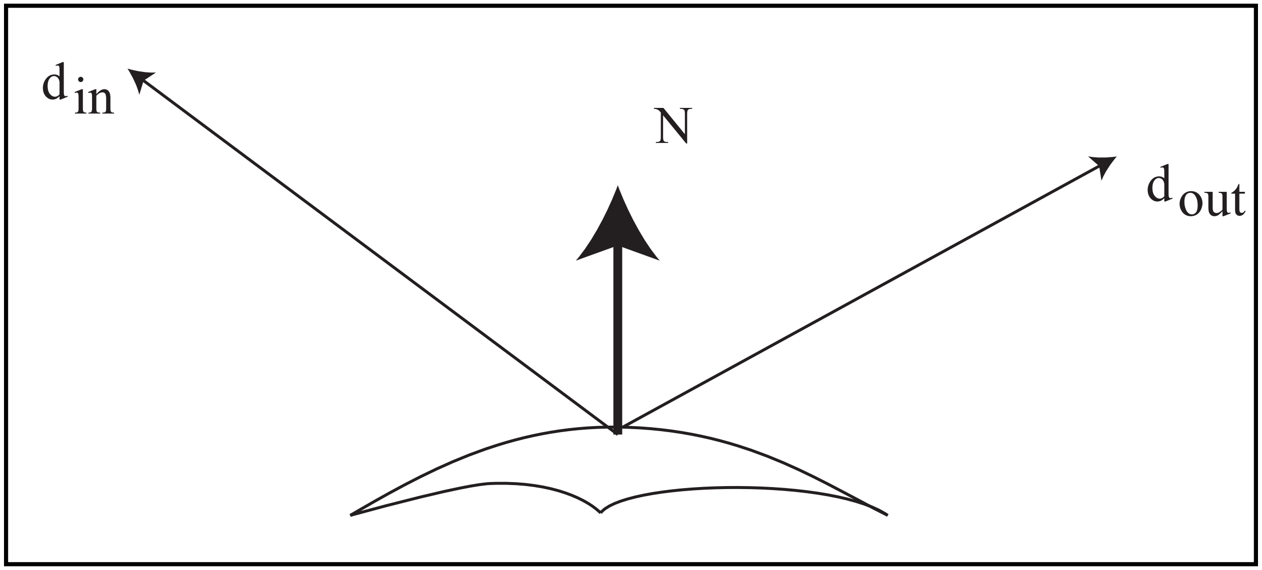


Diffuse reflection

- Light leaves the surface evenly in all directions
  - cotton cloth, carpets, matte paper, matte paints, etc.
  - most “rough” surfaces
  - Parameter: Albedo
    - percentage of light arriving that leaves
    - range 0-1
      - practical range is smaller
- Test:
  - surface has same apparent brightness when viewed from different dir’ns

Specular surface

- For some surfaces, reflection depends strongly on angle
  - mirrors (special case)
    - incoming direction, normal and outgoing direction are coplanar
    - angle din, normal and angle dout, normal are the same
  - more general cases later
  - rules:
    - din, dout, N coplanar
    - angle(din, N)=angle(dout, N)

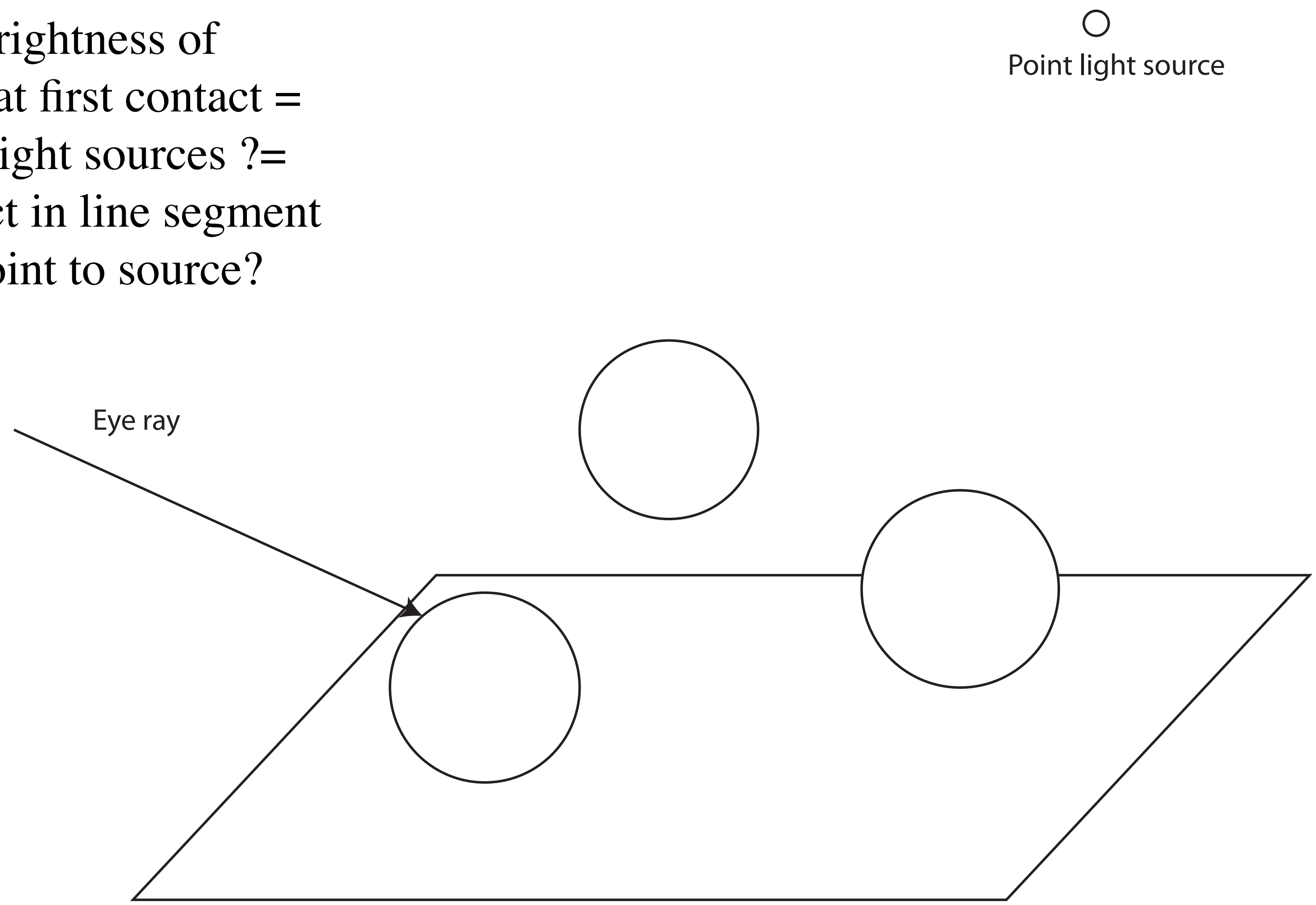


Lighting model

- Light arrives at a surface ONLY from a luminaire
  - this is an object that “makes light”
    - through chemical, mechanical, etc means
- Wild oversimplification, good for us right now
  - wait a few slides and it’ll get more complicated

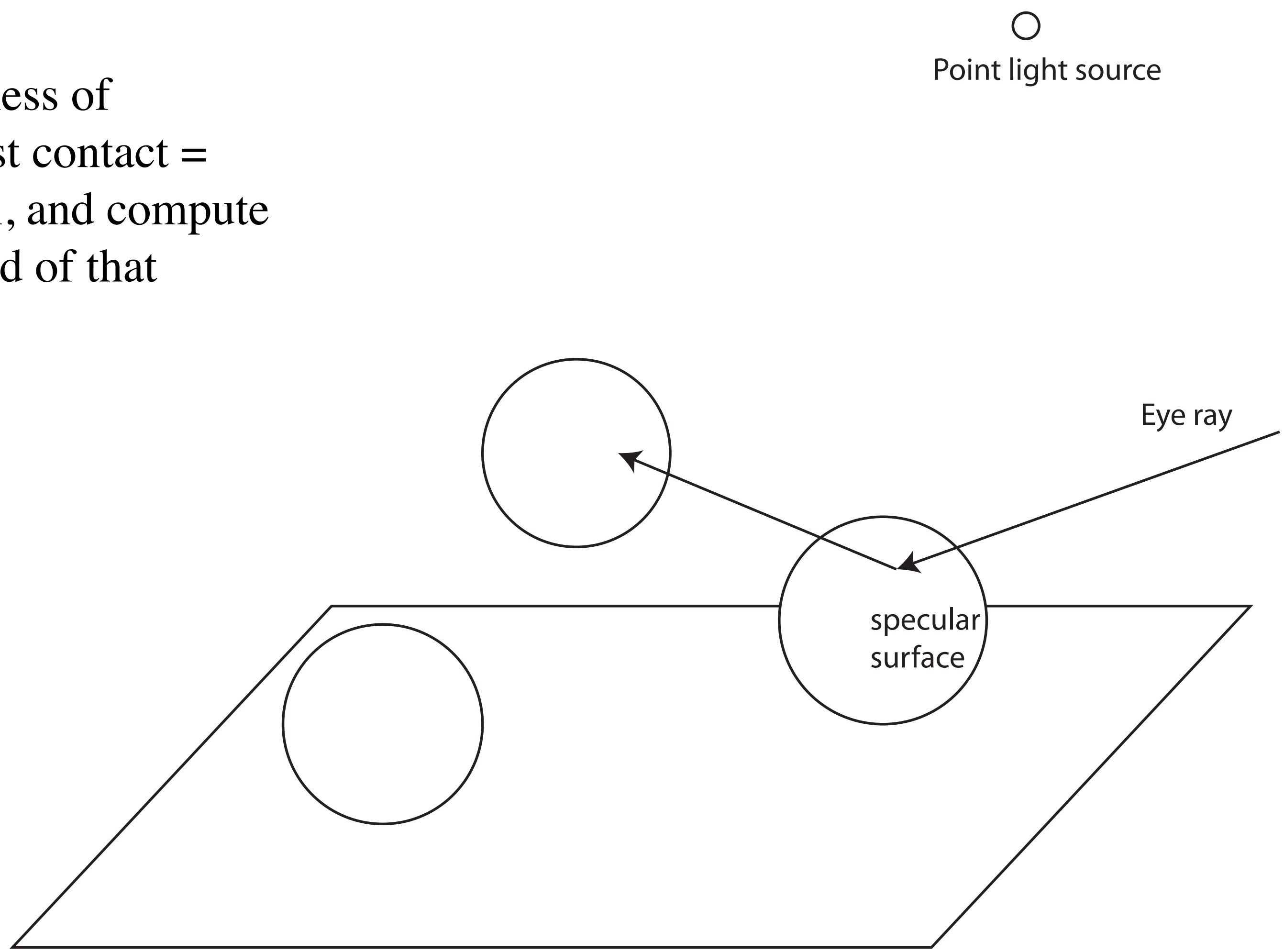
Eye ray strikes diffuse surface

Compute brightness of
diffuse surface at first contact =
Can it see the light sources ?=
Is there an object in line segment
connecting point to source?



Eye ray strikes specular surface

Compute brightness of
specular surface at first contact =
eye ray changes direction, and compute
brightness at the end of that



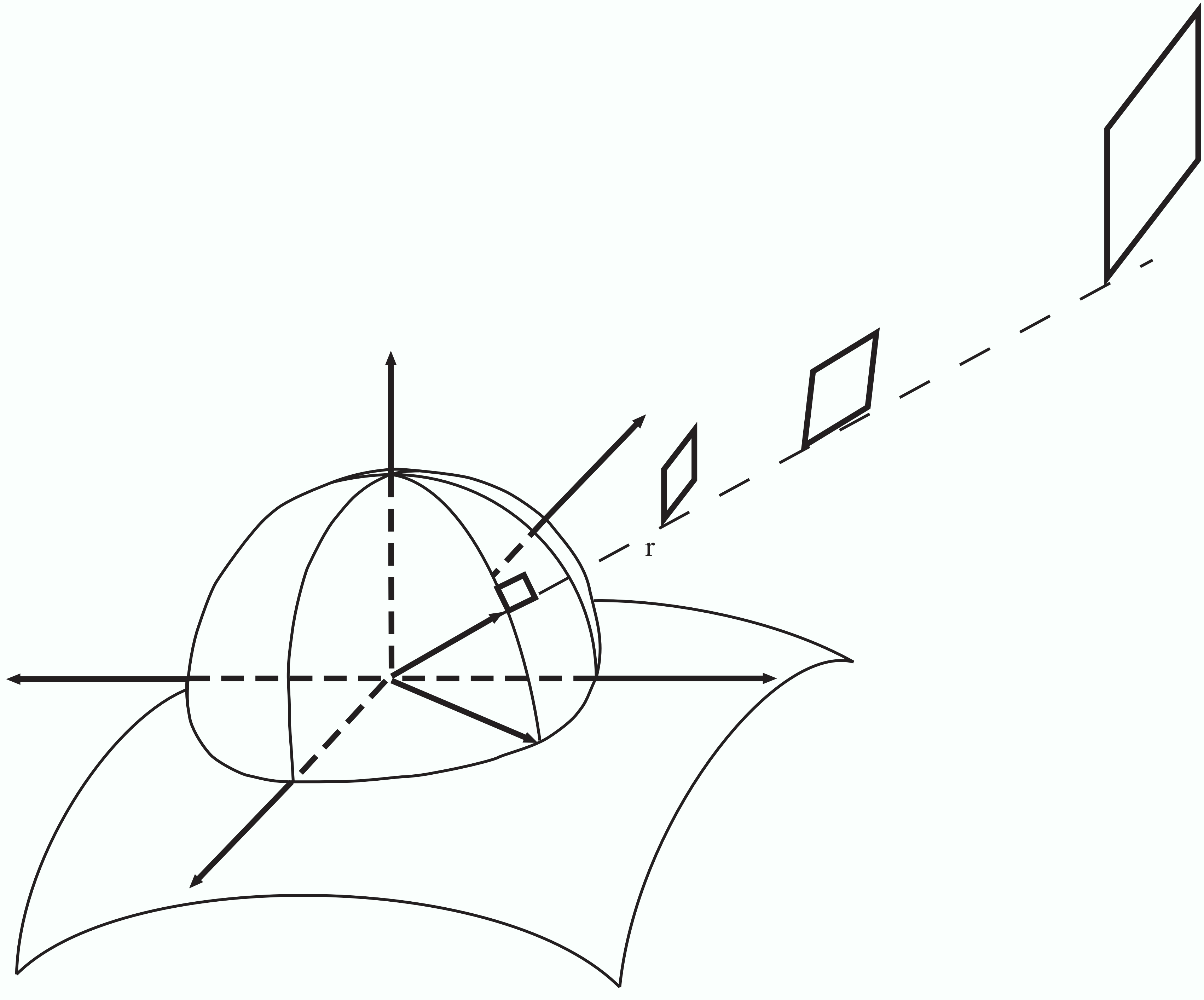
Implied computational problems

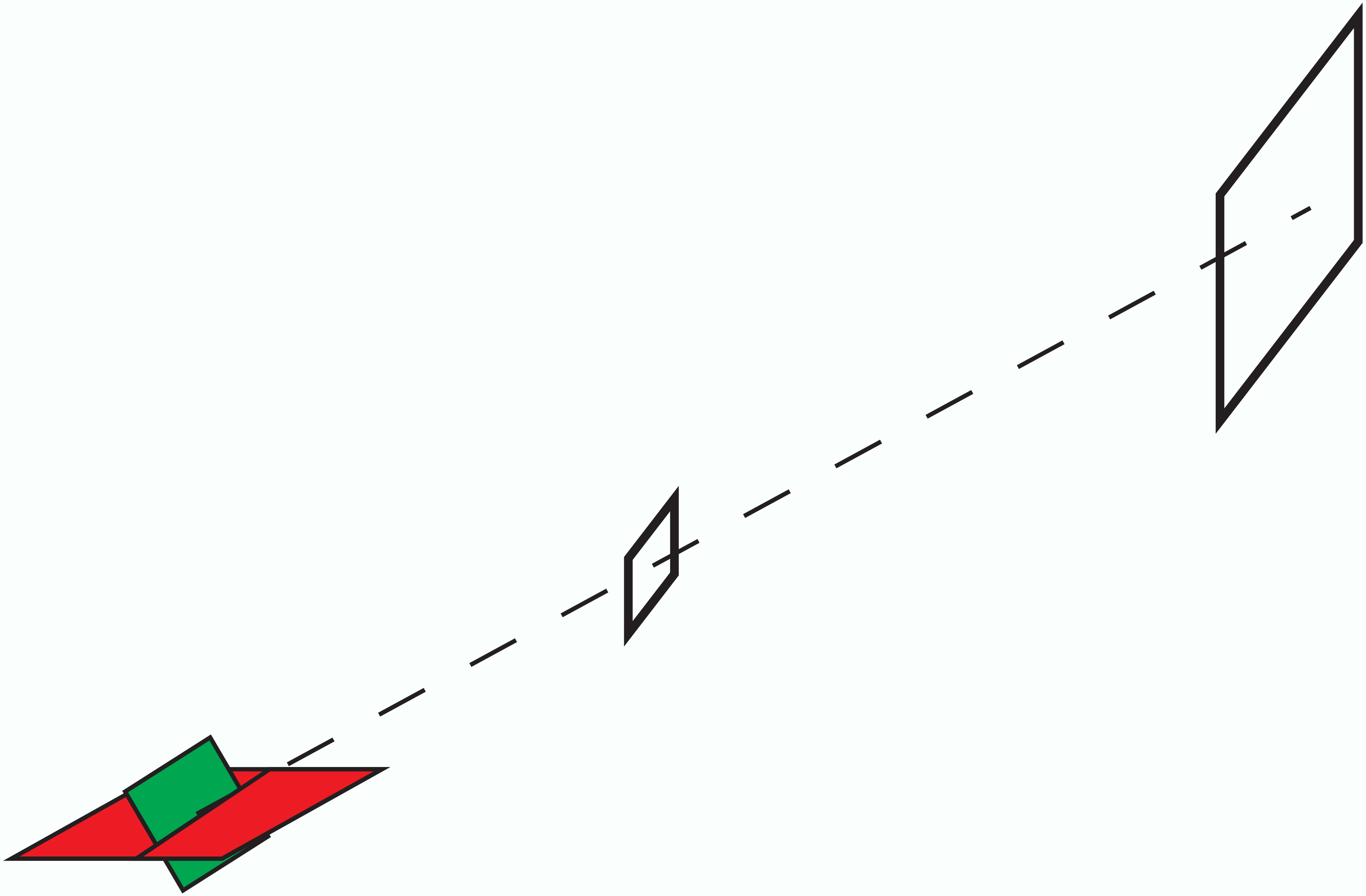
- Fast, accurate intersection with complicated models
  - Improved rendering
    - anti aliasing (= more rays)
    - motion blur (= more rays)
    - more complex illumination phenomena (= more rays, caching)

Key idea - how bright is this point?

Radiometry

- Questions:
  - how “bright” will surfaces be?
  - what is “brightness”?
    - measuring light
    - interactions between light and surfaces
- Core idea - think about light arriving at a surface
  - around any point is a hemisphere of directions
  - what is important is what a source “looks like” to a receiver
    - receiver can’t know anything else about source


r



Solid angle

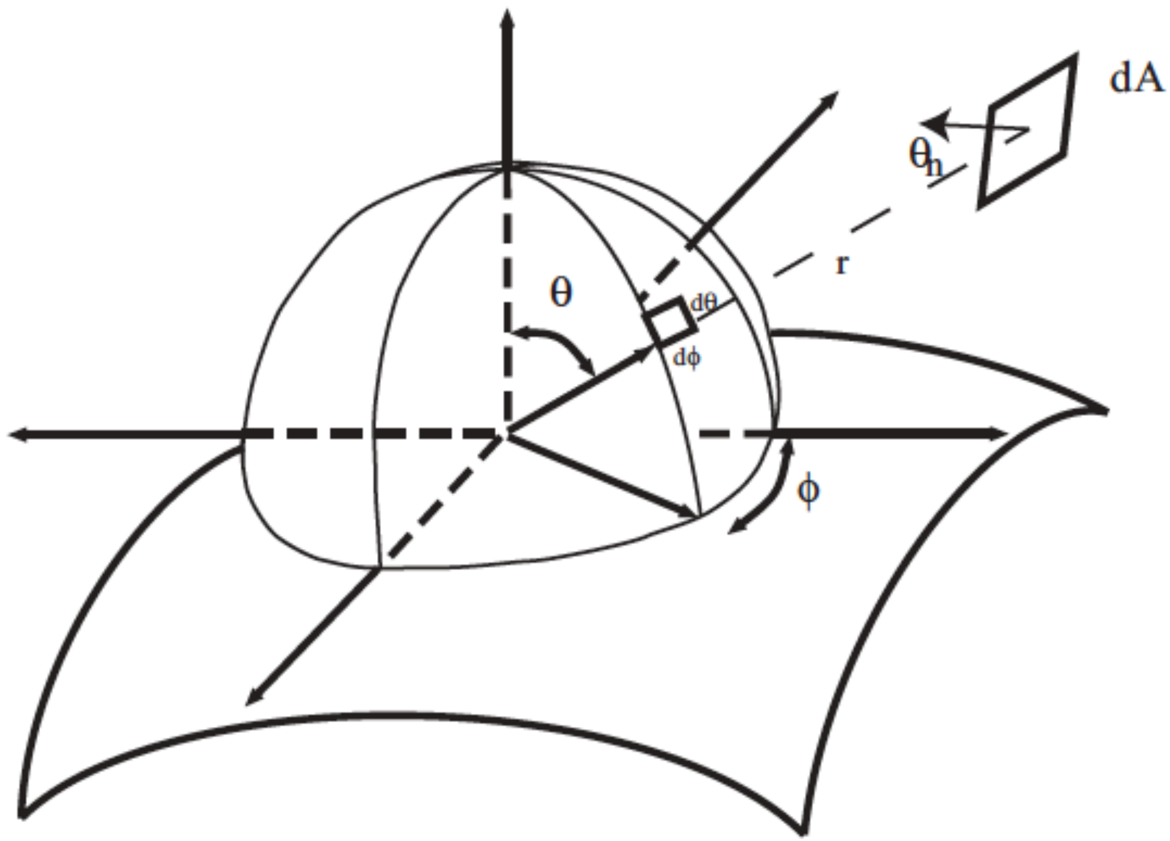


FIGURE 2.15: A hemisphere on a patch of surface, to show our angular coordinates for computing radiometric quantities. The coordinate axes are there to help you see the drawing as a 3D surface. An infinitesimal patch of surface with area dA which is distance r away is projected onto the unit hemisphere centered at the relevant point; the resulting area is the solid angle of the patch, marked as $d\theta d\phi$. In this case, the patch is small so that the area and hence the solid angle is $(1/r^2)dA\cos\theta_n$, where θ_n is the angle of inclination of the patch.

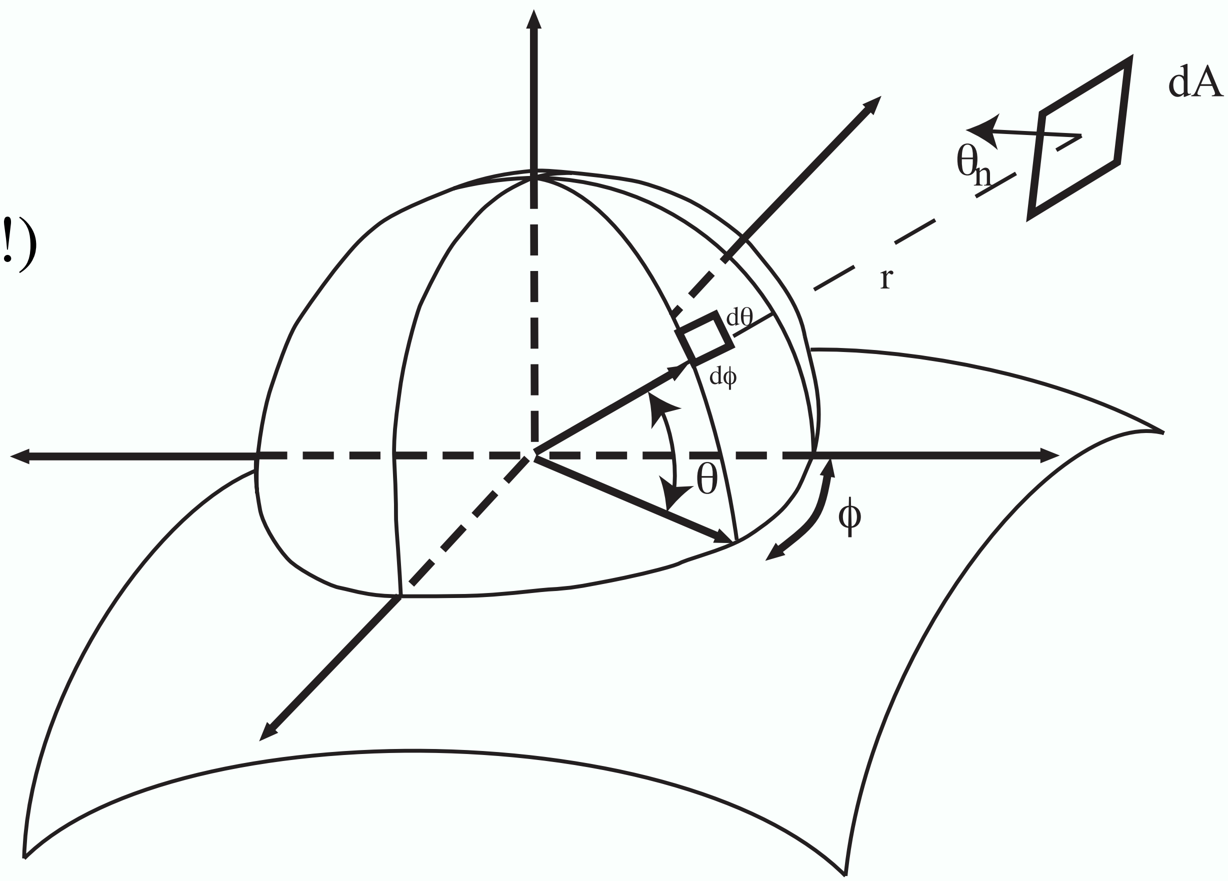
Solid Angle

- By analogy with angle (in radians)
- The solid angle subtended by a patch area dA is given by

$$d\omega = \frac{\cos\theta_n}{r^2} dA$$

- and (in right coords!)

$$d\omega = \cos\theta d\theta d\phi$$



Radiance

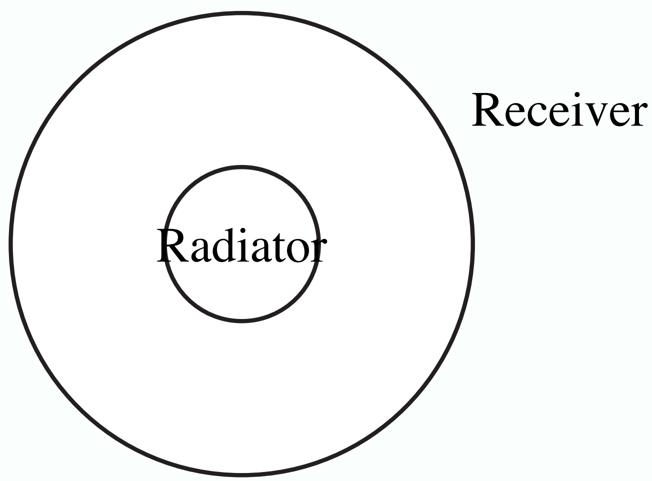
- Measure the "amount of light" at a point, in a direction

  the power (amount of energy per unit time) traveling at some point in a specified direction, per unit area *perpendicular to the direction of travel*, per unit solid angle.

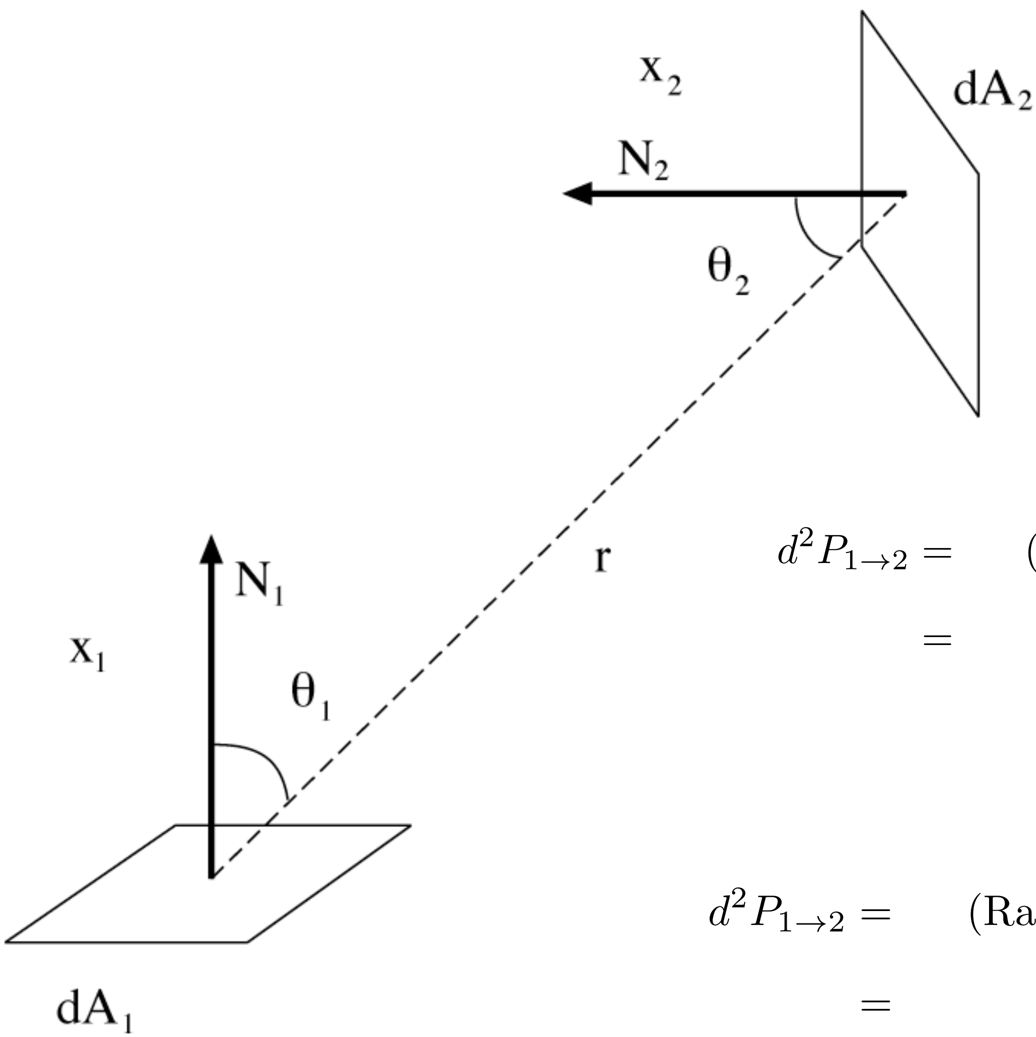
- Units: watts per square meter per steradian (wm-2sr-1)
- Crucial property:
  - In a vacuum, radiance leaving p in the direction of q is the same as radiance arriving at q from p
  - hence the units

Why not watts/square meter?

- Consider sphere radiating 1 W into vacuum
  - Radius 1, center at origin
  - Vacuum neither creates nor consumes power
- There's another sphere around it
  - Radius R, center at origin
  - Area - 4 pi R^2
  - It can't collect more power than first sphere radiates so
    - watts/square meter must go down with distance….!!! (ew)



Radiance is constant along straight lines



Power 1-> 2, leaving 1

$$d^2 P_{1\to 2} = \quad \text{(Radiance)(foreshortened area of 1)(solid angle of 2 at 1)}$$

$$= \quad L(\mathbf{x}_1, \mathbf{x}_1 \to \mathbf{x}_2)(\cos\theta_1 dA_1)(\frac{\cos\theta_2}{r^2} dA_2)$$

Power 1-> 2, arriving at 2

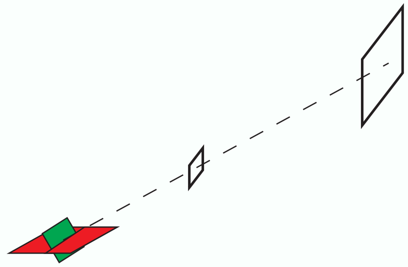
$$d^2 P_{1\to 2} = \quad \text{(Radiance)(foreshortened area of 2)(solid angle of 1 at 2)}$$

$$= \quad L(\mathbf{x}_2, \mathbf{x}_1 \to \mathbf{x}_2)(\cos\theta_2 dA_2)(\frac{\cos\theta_1}{r^2} dA_1)$$

Irradiance

- How much light is arriving at a surface?
- Sensible unit is Irradiance
  - Incident power per unit area not foreshortened
  - This is a function of incoming angle.
- A surface experiencing radiance $L(x,\theta,\phi)$ coming in from $d\omega$ experiences irradiance

$$L(\mathbf{x}, \theta, \phi) \cos\theta d\omega$$



• Crucial property:
Total power arriving at the surface is given by adding irradiance over all incoming angles --- this is why it's a natural unit

Surfaces and the BRDF

- Many effects when light strikes a surface -- could be:
  - absorbed; transmitted. reflected; scattered
- Assume that
  - surfaces don’t fluoresce
  - surfaces don’t emit light (i.e. are cool)
  - all the light leaving a point is due to that arriving at that point
- Can model this situation with the Bidirectional Reflectance Distribution Function (BRDF)
- the ratio of the radiance in the outgoing direction to the incident irradiance

$$\rho_{bd}\left(\underline{x},\vartheta_o,\varphi_o,\vartheta_i,\varphi_i,\right) = \frac{L_o\left(\underline{x},\vartheta_o,\varphi_o\right)}{L_i\left(\underline{x},\vartheta_i,\varphi_i\right)\cos\vartheta_i d\omega}$$

BRDF

- Units: inverse steradians (sr-1)
- Symmetric in incoming and outgoing directions
- Radiance leaving in a particular direction:
  - add contributions from every incoming direction

$$\int_{\Omega} \rho_{bd}\left(\underline{x}, \vartheta_o, \varphi_o, \vartheta_i, \varphi_i,\right) L_i\left(\underline{x}, \vartheta_i, \varphi_i\right) \cos \vartheta_i \, d\omega_i$$

Suppressing Angles - Radiosity

- In many situations, we do not really need angle coordinates
  - e.g. cotton cloth, where the reflected light is not dependent on angle
- Appropriate radiometric unit is radiosity
  - total power leaving a point on the surface, per unit area on the surface (Wm-2)
- Radiosity from radiance?
  - sum radiance leaving surface over all exit directions

$$B(\underline{x}) = \int_{\Omega} L_o(\underline{x}, \vartheta, \varphi) \cos\vartheta \, d\omega$$

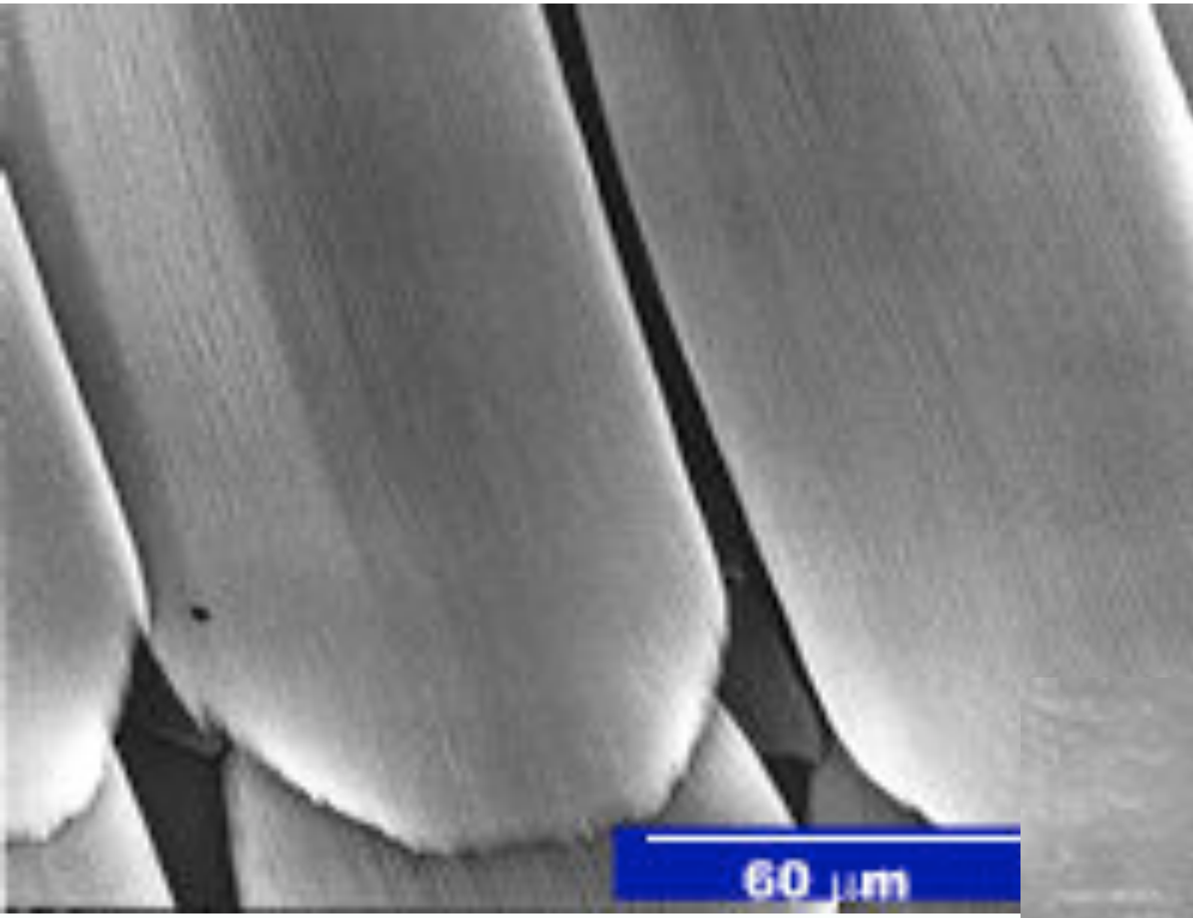
Exitance

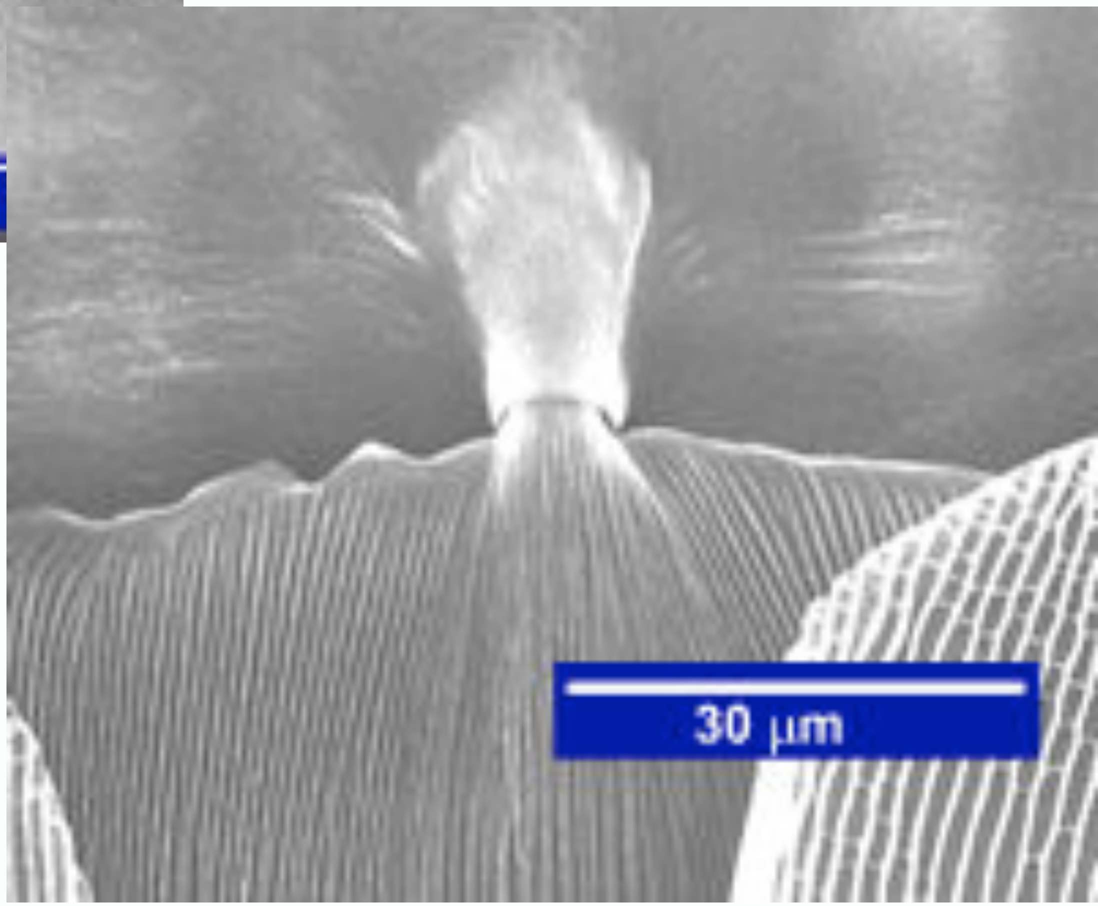
- For some luminaires, generated light independent of angle
  - think light box
- Appropriate radiometric unit is exitance
  - total power leaving a point on the surface, per unit area on the surface (Wm-2), created in the surface

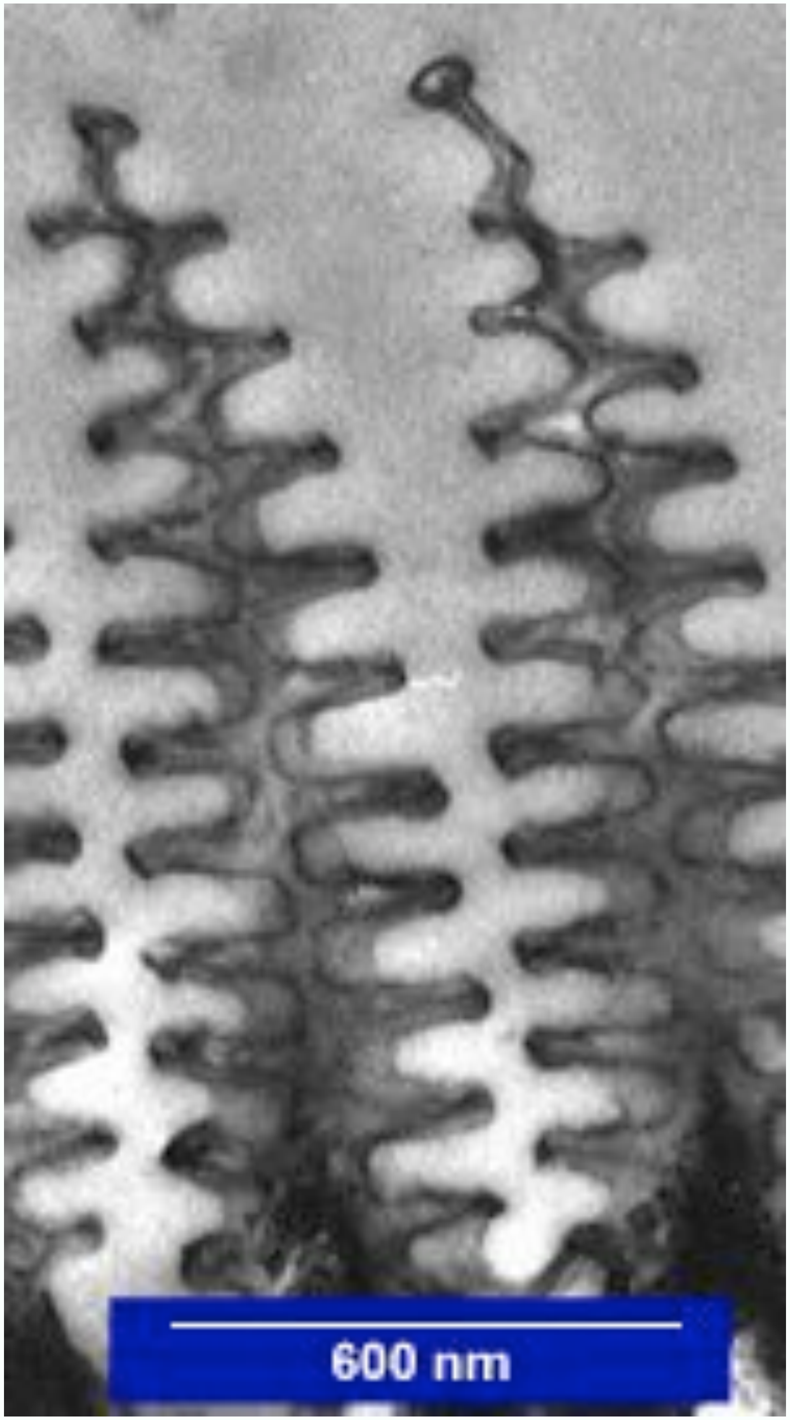
Radiosity

- Important relationship:
  - radiosity of a surface whose radiance is independent of angle (e.g. that cotton cloth)

$$
\begin{aligned}
B(\underline{x}) &= \int_{\Omega} L_o(\underline{x},\vartheta,\varphi)\cos\vartheta\, d\omega \\
&= L_o(\underline{x}) \int_{\Omega} \cos\vartheta\, d\omega \\
&= L_o(\underline{x}) \int_0^{\pi/2}\int_0^{2\pi} \cos\vartheta \sin\vartheta\, d\varphi\, d\vartheta \\
&= \pi L_o(\underline{x})
\end{aligned}
$$


60 µm


30 µm


600 nm

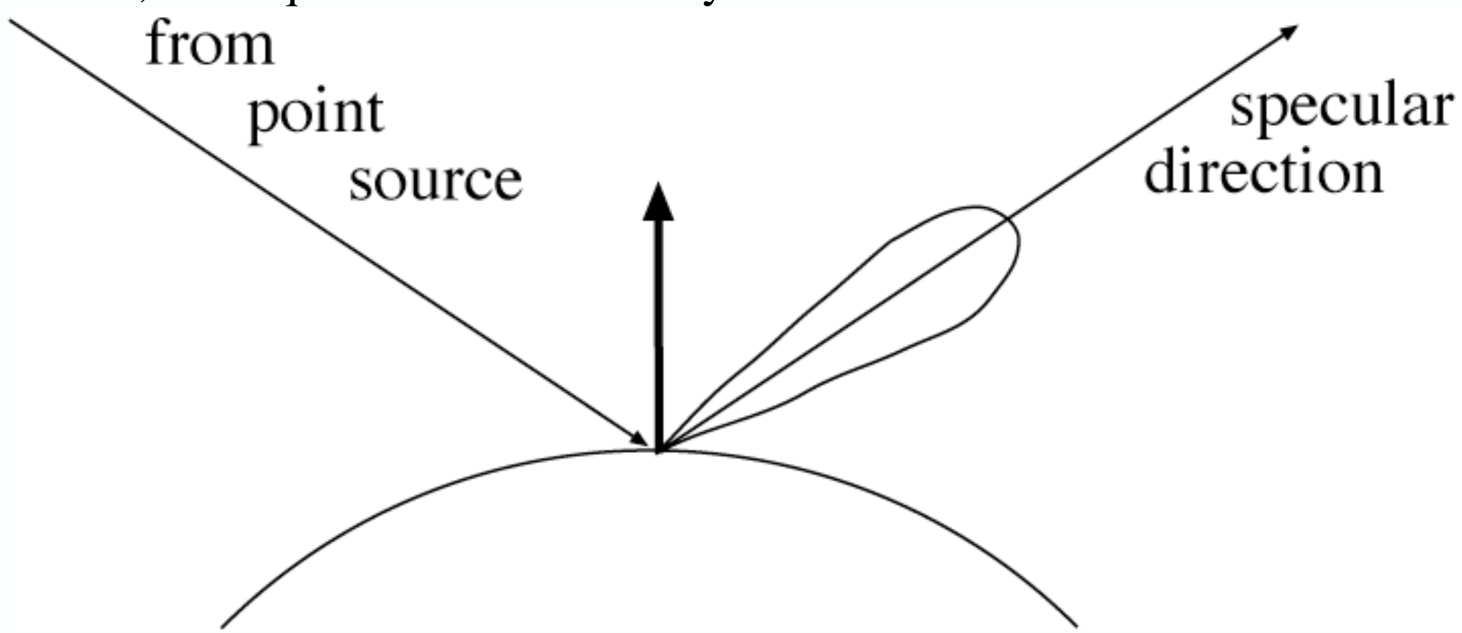
Lambertian surfaces and albedo

- For some surfaces, the BRDF is independent of direction
  - cotton cloth, carpets, matte paper, matte paints, etc.
  - radiance leaving the surface is independent of angle
  - Lambertian surfaces (same Lambert) or ideal diffuse surfaces
  - Use radiosity as a unit to describe light leaving the surface
  - percentage of incident light reflected is diffuse reflectance or albedo
- Useful fact:

$$\rho_{brdf} = \frac{\rho_d}{\pi}$$

Specular surfaces

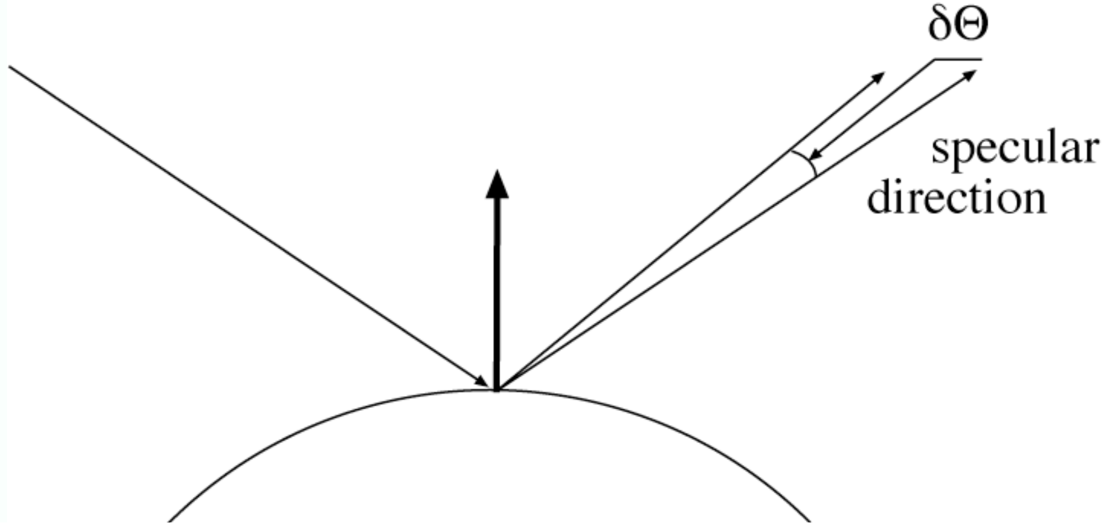
- Another important class of surfaces is specular, or mirror-like.
  - radiation arriving along a direction leaves along the specular direction
  - reflect about normal
  - some fraction is absorbed, some reflected
  - on real surfaces, energy usually goes into a lobe of directions
  - can write a BRDF, but requires the use of funny functions



Phong's model

- There are very few cases where the exact shape of the specular lobe matters.
- Typically:
  - very, very small --- mirror
  - small -- blurry mirror
  - bigger -- see only light sources as "specularities"
  - very big -- faint specularities
- Phong's model
  - reflected energy falls off with

$$\cos^n(\delta\vartheta)$$



Lambertian + specular

- Widespread model
  - all surfaces are Lambertian plus specular component
- Advantages
  - easy to manipulate
  - very often quite close true
- Disadvantages
  - some surfaces are not
    - e.g. underside of CD's, feathers of many birds, blue spots on many marine crustaceans and fish, most rough surfaces, oil films (skin!), wet surfaces
  - Generally, very little advantage in modelling behaviour of light at a surface in more detail -- it is quite difficult to understand behaviour of L+S surfaces

The Rendering Equation- 1

- We can now write

$$L_o(\mathbf{x}, \omega_o) = L_e(\mathbf{x}, \omega_o) + \int_\Omega \rho_{bd}(\mathbf{x}, \omega_o, \omega_i) L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i$$

Angle between normal and incoming direction

BRDF

Incoming radiance

Average over hemisphere

Radiance emitted from surface at that point in that direction

Radiance leaving a point in a direction

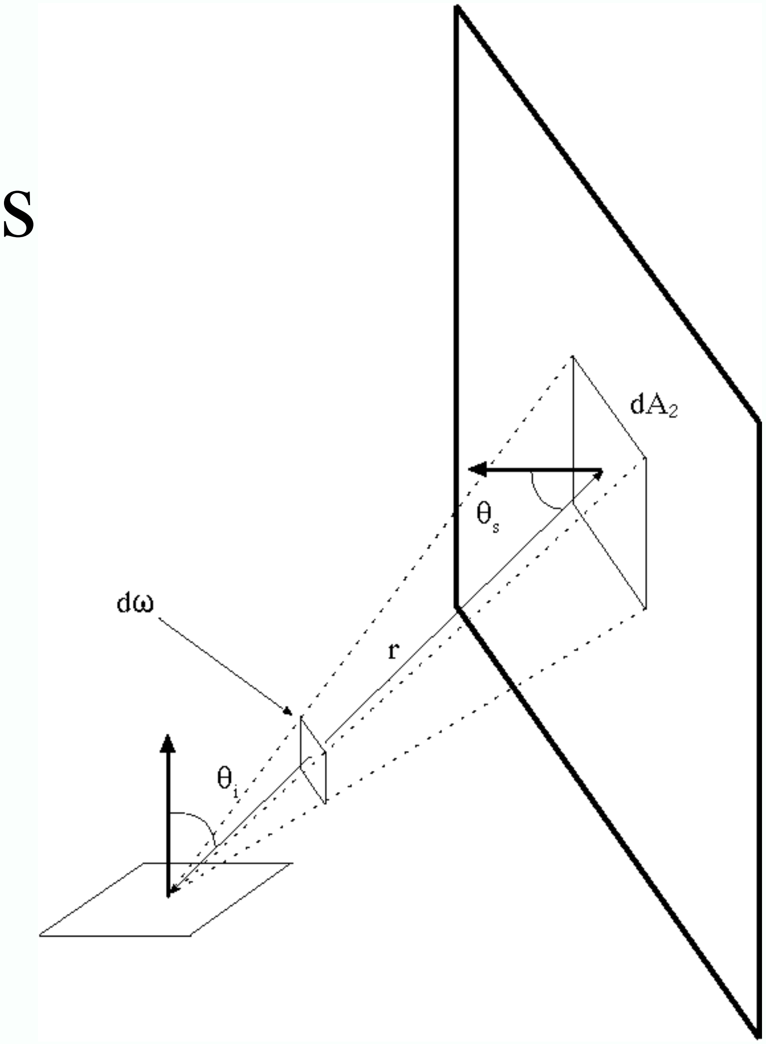
Radiance is constant along straight lines, so this is what we want to know

The Rendering Equation - II

- This balance works for
  - each wavelength,
  - at any time, so
- So

$$
\begin{aligned}
L_o(\mathbf{x}, \omega_o, \lambda, t) &= L_e(\mathbf{x}, \omega_o, \lambda, t) + \\
&\int_\Omega \rho_{bd}(\mathbf{x}, \omega_o, \omega_i, \lambda, t) L_i(\mathbf{x}, \omega_i, \lambda, t) \cos\theta_i d\omega_i
\end{aligned}
$$

Area sources



- Examples: diffuser boxes, white walls.
- The radiosity at a point due to an area source is obtained by adding up the contribution over the section of view hemisphere subtended by the source
  - change variables and add up over the source

Radiosity due to an area source

$$B(x) = \rho_d(x)\int_{\Omega} L_i(x, u \to x)\cos\theta_i \, d\omega$$

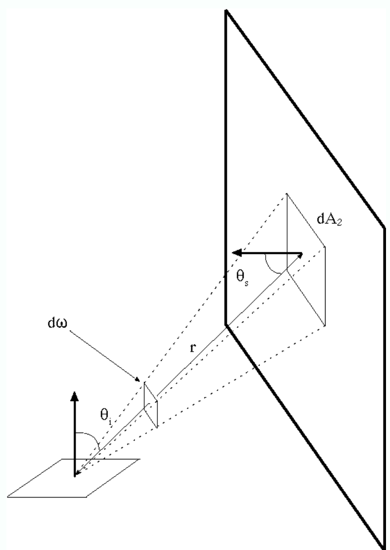
$$= \rho_d(x)\int_{\Omega} L_e(x, u \to x)\cos\theta_i \, d\omega$$

$$= \rho_d(x)\int_{\Omega} \left(\frac{E(u)}{\pi}\right)\cos\theta_i \, d\omega$$

$$= \rho_d(x)\int_{source} \left(\frac{E(u)}{\pi}\right)\cos\theta_i \left(\cos\theta_s \frac{dA_u}{r(x,u)^2}\right)$$

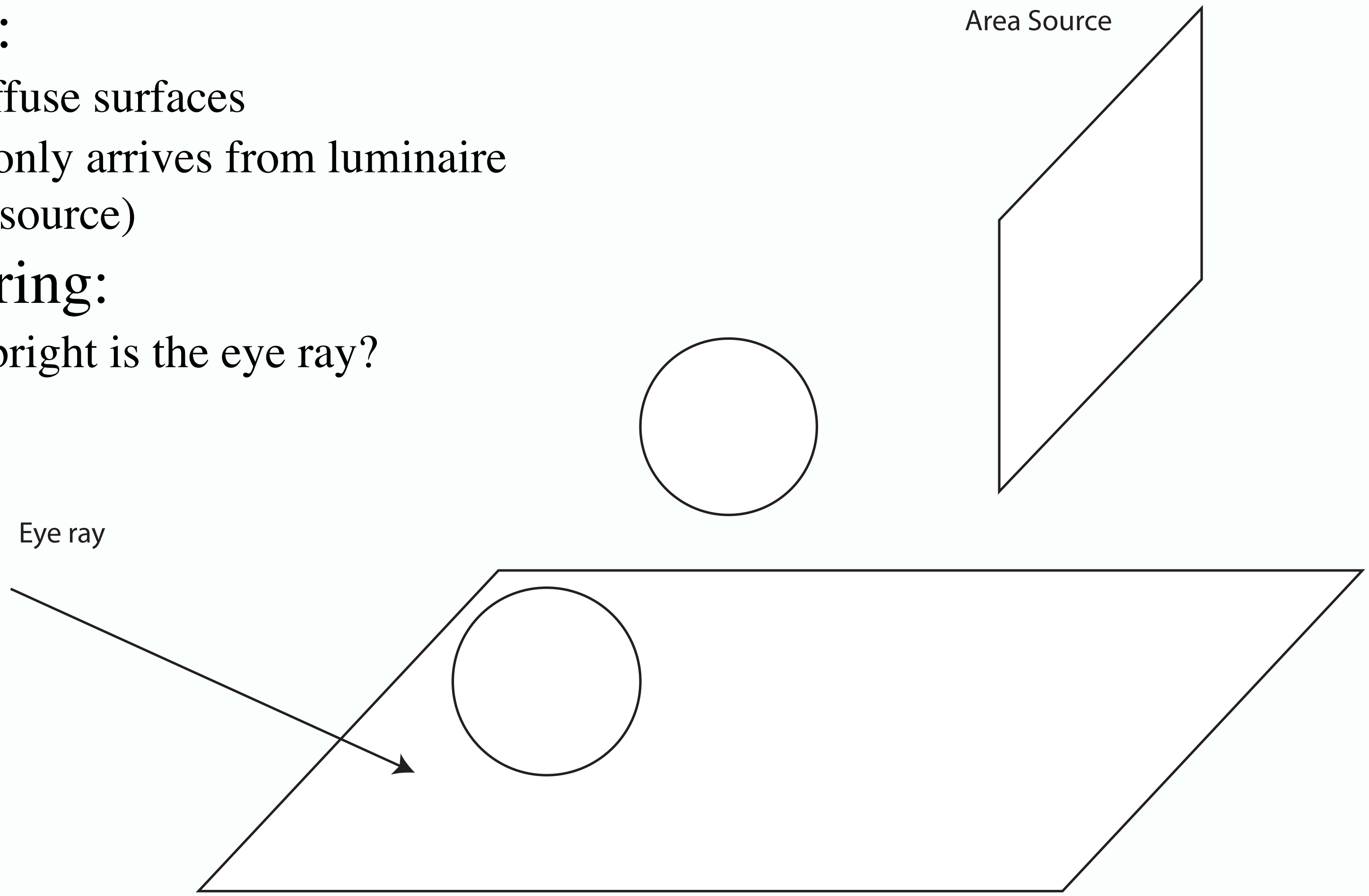
$$= \rho_d(x)\int_{source} E(u)\frac{\cos\theta_i \cos\theta_s}{\pi r(x,u)^2} dA_u$$

- rho is albedo
- E is exitance
- r(x, u) is distance between points
- u is a coordinate on the source



Question: how to ray-trace this?

- Model:
  - all diffuse surfaces
  - light only arrives from luminaire (area source)
- Rendering:
  - how bright is the eye ray?



Recall

Diffuse, so this is a constant

Angle between normal and incoming direction

$$L_o(\mathbf{x}, \omega_o) = L_e(\mathbf{x}, \omega_o) + \int_\Omega \rho_{bd}(\mathbf{x}, \omega_o, \omega_i) L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i$$

BRDF

Incoming radiance

This is from area source

Average over hemisphere

Radiance emitted from surface at that point in that direction

There isn't any, so zero

Radiance leaving a point in a direction

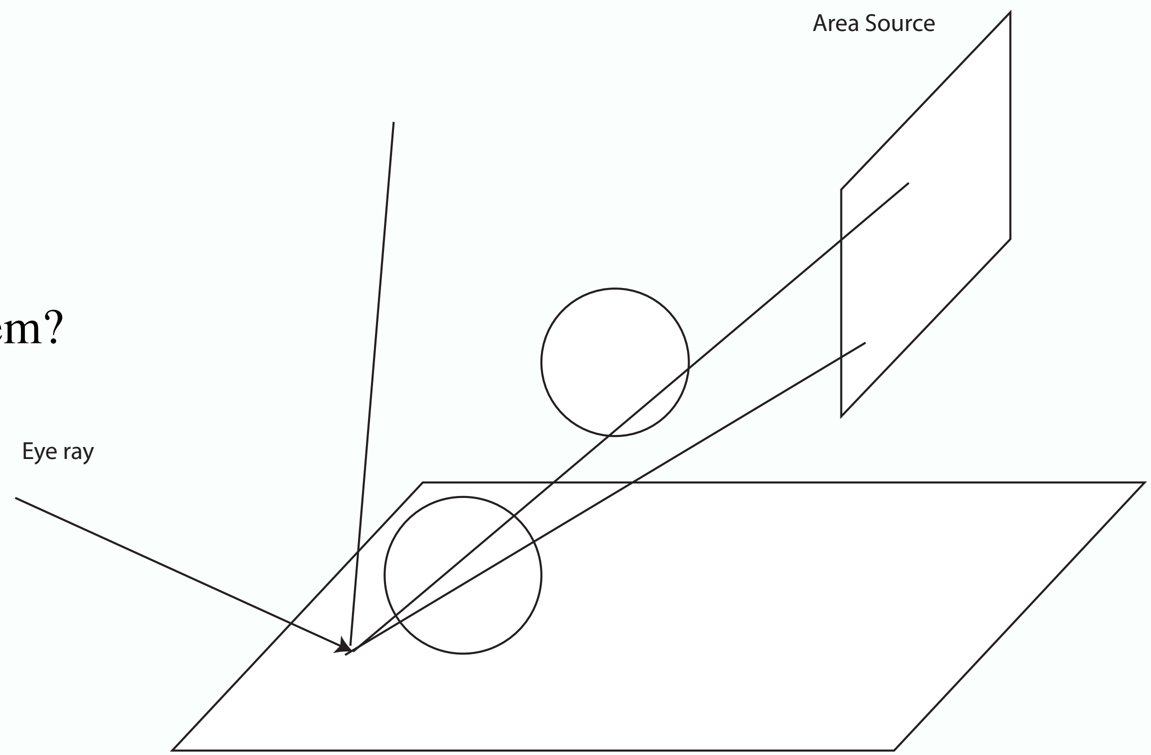
Radiance along eye ray

$$L_o(\mathbf{x},\omega_o) = L_e(\mathbf{x},\omega_o) + \int_\Omega \rho_{bd}(\mathbf{x},\omega_o,\omega_i) L_i(\mathbf{x},\omega_i)\cos\theta_i d\omega_i$$

0

$$\frac{1}{N} \sum_{\omega_i \in \text{samples of incoming directions}} \rho L(\mathbf{x},\omega_i)\cos\theta_i$$

But which directions?
and how should we sample them?

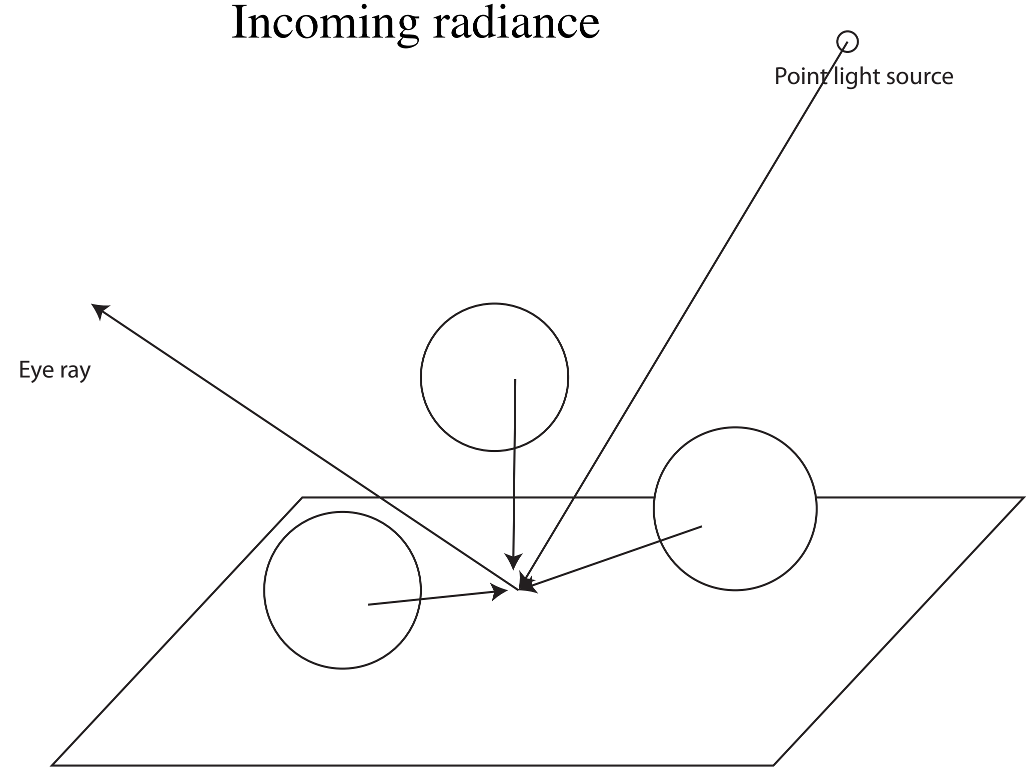


Global illumination

$$L_o(\mathbf{x}, \omega_o) = L_e(\mathbf{x}, \omega_o) + \int_\Omega \rho_{bd}(\mathbf{x}, \omega_o, \omega_i) L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i$$

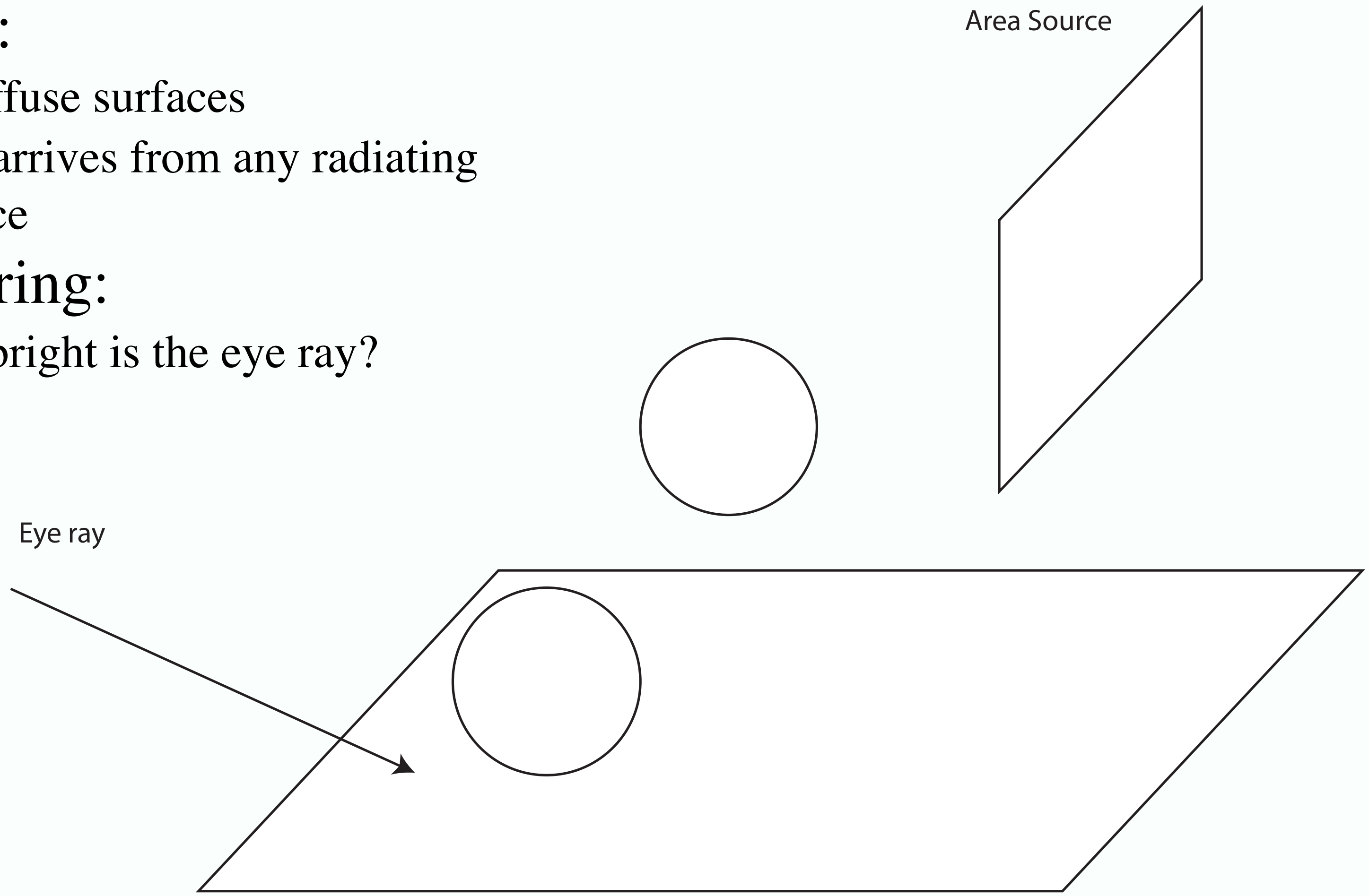
Incoming radiance

- Incoming radiance isn't just from luminaires
  - the reason you can see surfaces is they reflect light
  - other surfaces don't distinguish between reflected light and generated light



Question: how to ray-trace this?

- Model:
  - all diffuse surfaces
  - light arrives from any radiating surface
- Rendering:
  - how bright is the eye ray?



Recall

Diffuse, so this is a constant

Angle between normal and incoming direction

$$L_o(\mathbf{x}, \omega_o) = L_e(\mathbf{x}, \omega_o) + \int_{\Omega} \rho_{bd}(\mathbf{x}, \omega_o, \omega_i) L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i$$

BRDF

Incoming radiance

This is now from any radiator

Average over hemisphere

Radiance emitted from surface at that point in that direction

There isn't any, so zero

Radiance leaving a point in a direction

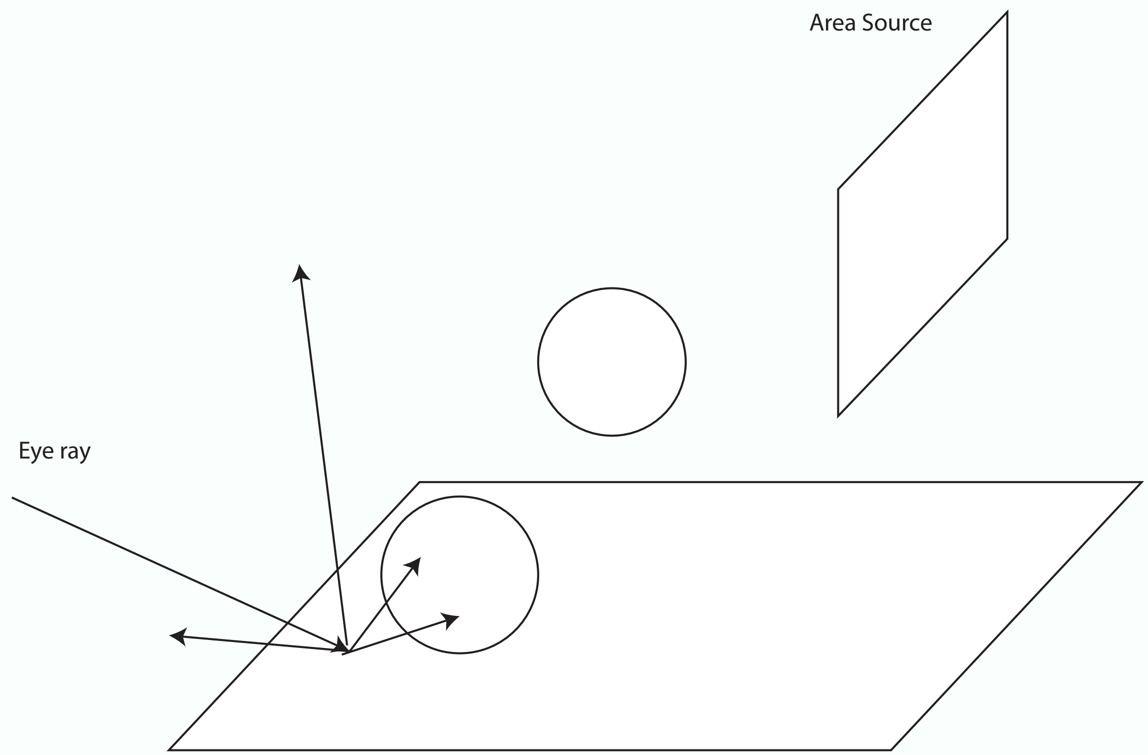
Radiance along eye ray

$$L_o(\mathbf{x}, \omega_o) = L_e(\mathbf{x}, \omega_o) + \int_\Omega \rho_{bd}(\mathbf{x}, \omega_o, \omega_i) L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i$$

0

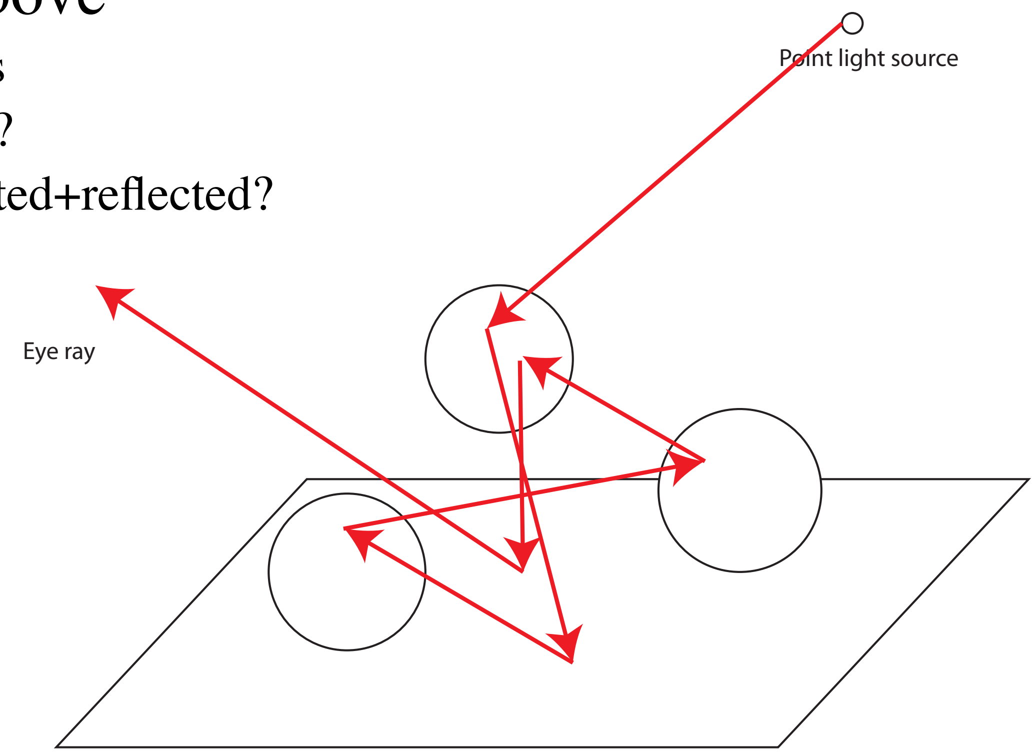
$$\frac{1}{N} \sum_{\omega_i \in \text{samples of incoming directions}} \rho L(\mathbf{x}, \omega_i) \cos\theta_i$$

But which directions?
and how should we sample them?



Light paths

- Recursively expand, as above
  - sample the incoming directions
    - what radiance is coming in?
    - go to far end - what is emitted+reflected?
    - recur
-



Light paths - II

$$v = \int_\Lambda \int_D \int_\Omega \int_T w(\mathbf{x}, \lambda, \omega, t) L(\mathbf{x}, \omega, t) dt d\omega dx d\lambda \approx \sum_{i \in \text{rays}} g(\text{ray}) L(\text{ray}$$

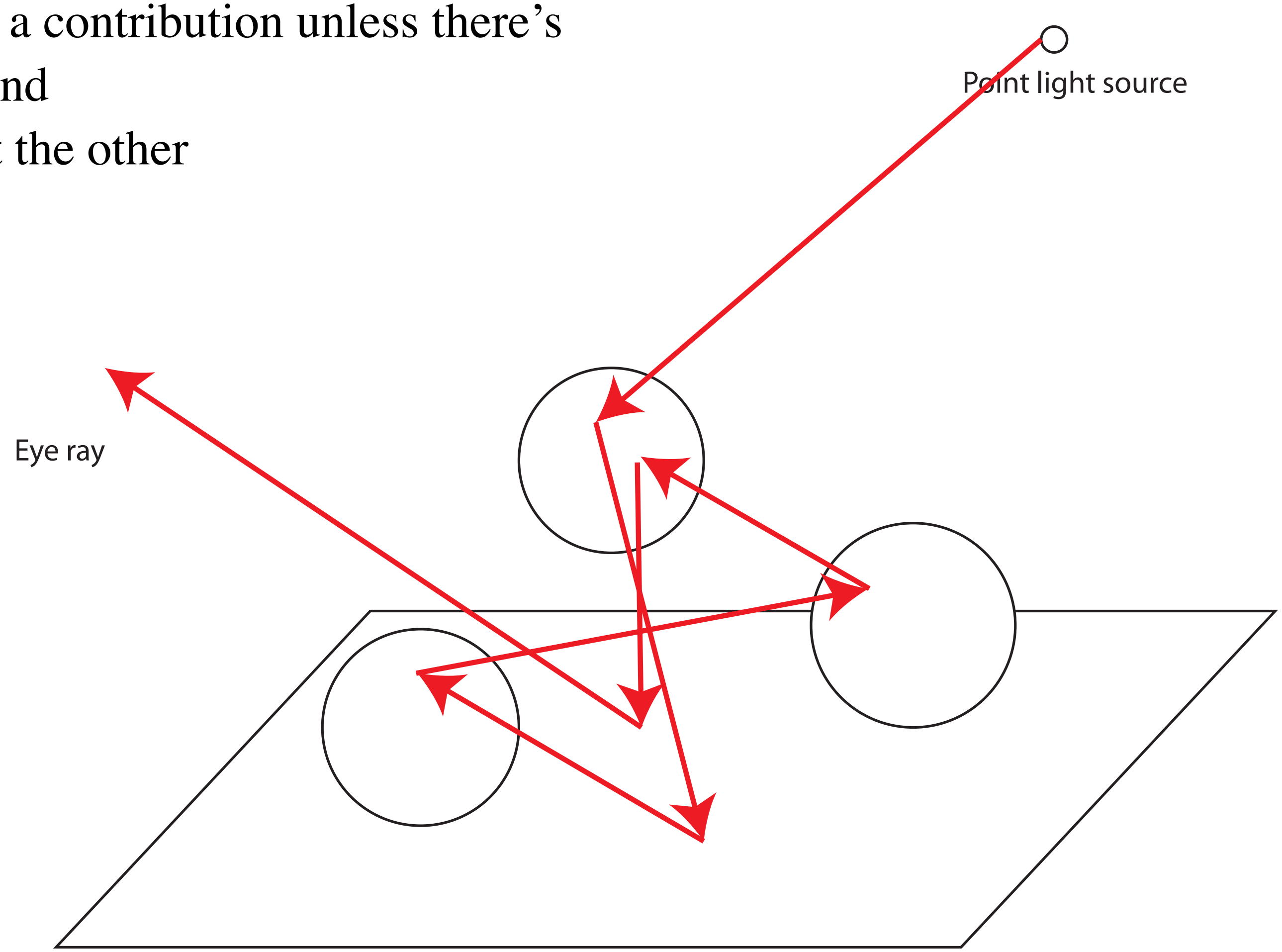
- But this is really (suppressing wavelength and time)

$$\int_D \int_\Omega L(\mathbf{x}, \omega) w(\mathbf{x}, \omega) dx d\omega \approx \frac{1}{N} \sum_{paths} (\text{contribution of path})$$

Light paths - III

- Now consider contribution of path
  - it doesn't make a contribution unless there's
    - eye at one end
    - luminaire at the other
- We can write

L (Something) E

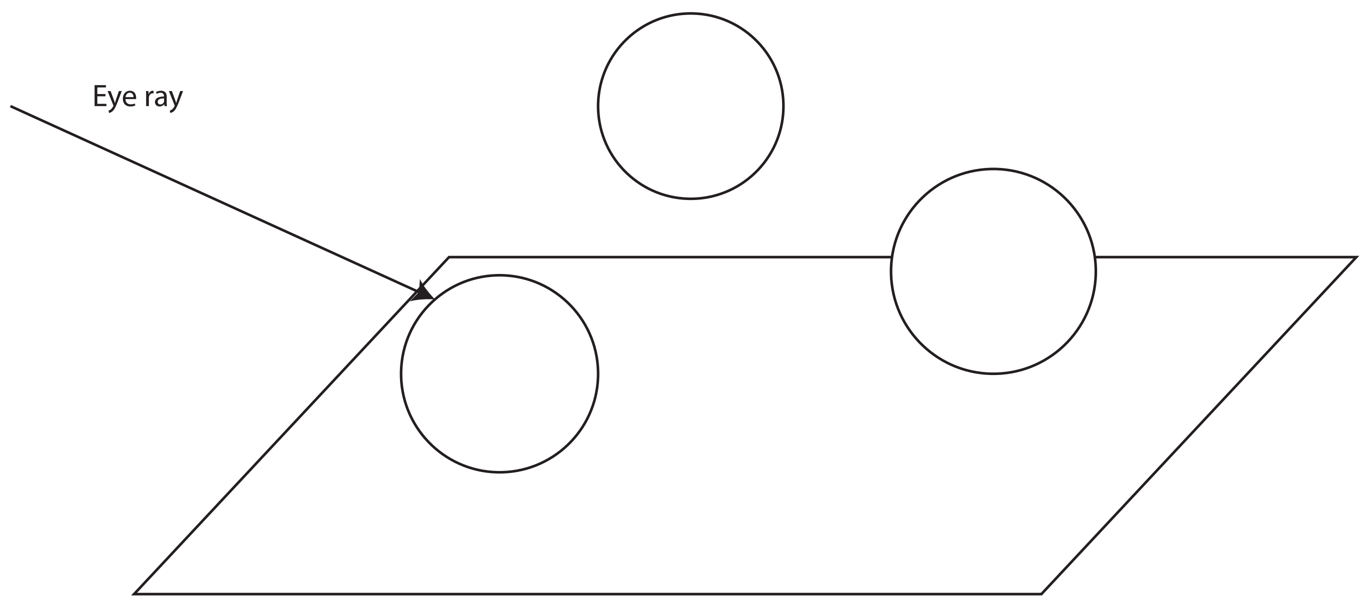


Some light paths are harder than others

- We have already seen how to render
  - LDE - (light diffuse eye)
    - eye ray to diffuse surface, can it see light?
  - LSE - (light specular eye)
    - eye ray to specular surface, reflect and hit diffuse, can it see light?
  - Actually, can do:
    - LDS*E - (light diffuse 0 or more specular bounces eye)
- How about
  - LDDE - (light diffuse diffuse eye)
    - easy geometry likely high variance
  - LS+DE - (light diffuse at least one specular eye)
    - rather harder

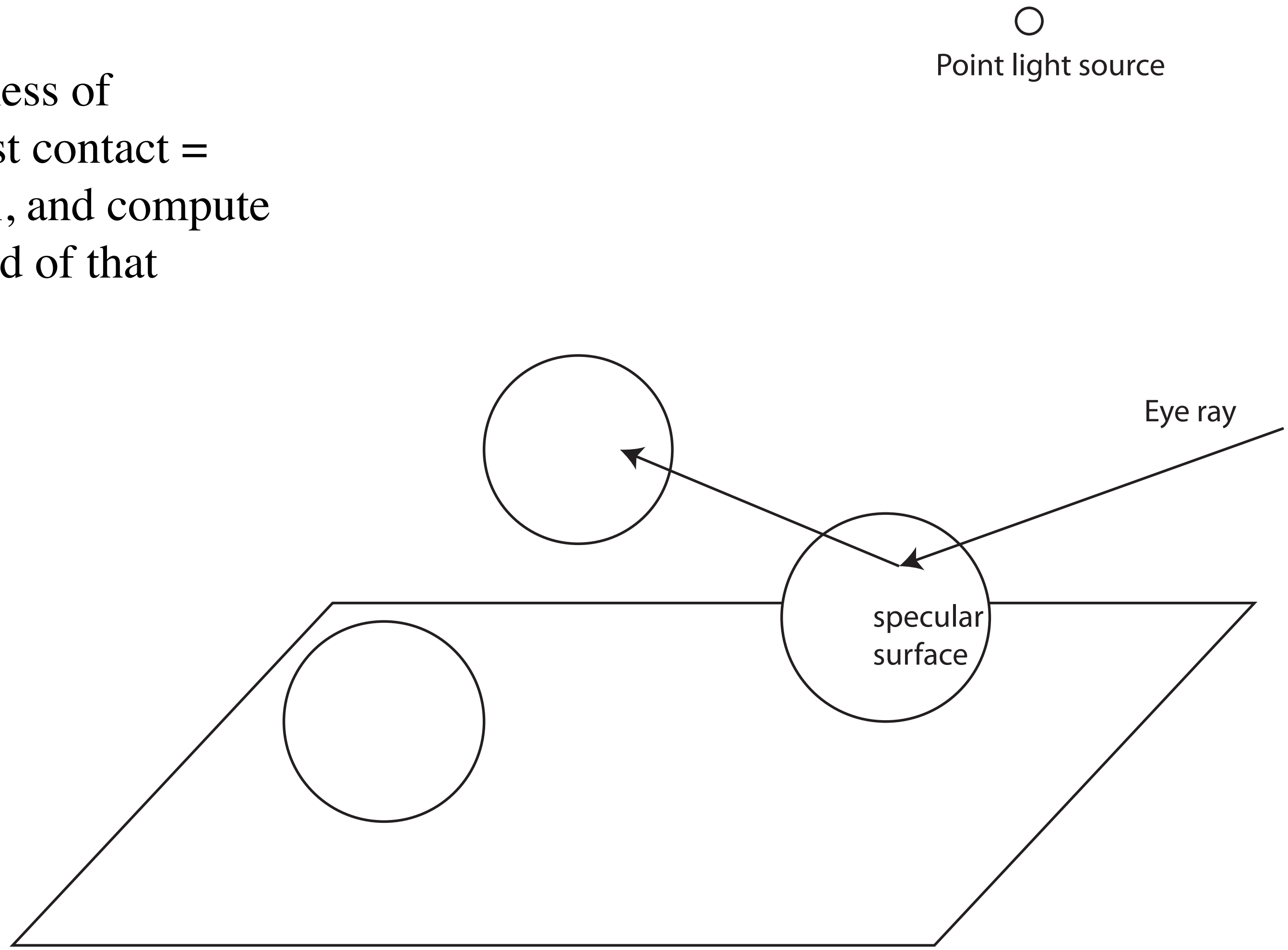
Eye ray strikes diffuse surface - LDE

Compute brightness of
diffuse surface at first contact =
Can it see the light sources ?=
Is there an object in line segment
connecting point to source?



Eye ray strikes specular surface - LDSE

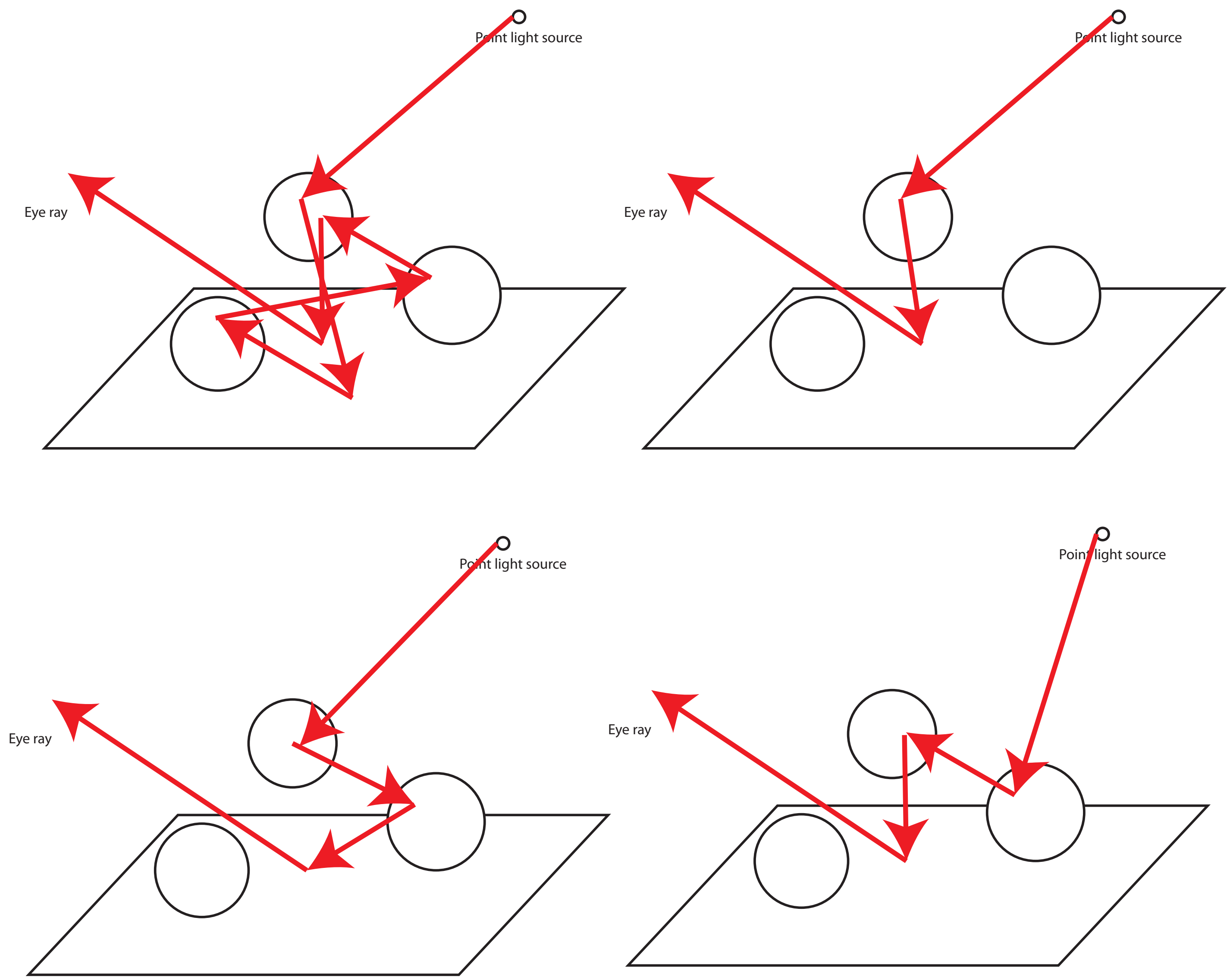
Compute brightness of
specular surface at first contact =
eye ray changes direction, and compute
brightness at the end of that



Diffuse VS Specular (also translucent)

- Diffuse surfaces:
  - any incoming direction can cause light to travel down the eye ray
    - so you do not know from which directions contributions will arrive
  - when an eye ray arrives, it must create multiple query rays
- Specular surfaces:
  - only one incoming direction can cause light to travel down the eye ray
    - so you do know from which directions contributions will arrive
  - when an eye ray arrives, it creates only one query ray
- Translucent surfaces are like specular surfaces:
  - different geometry for the query ray

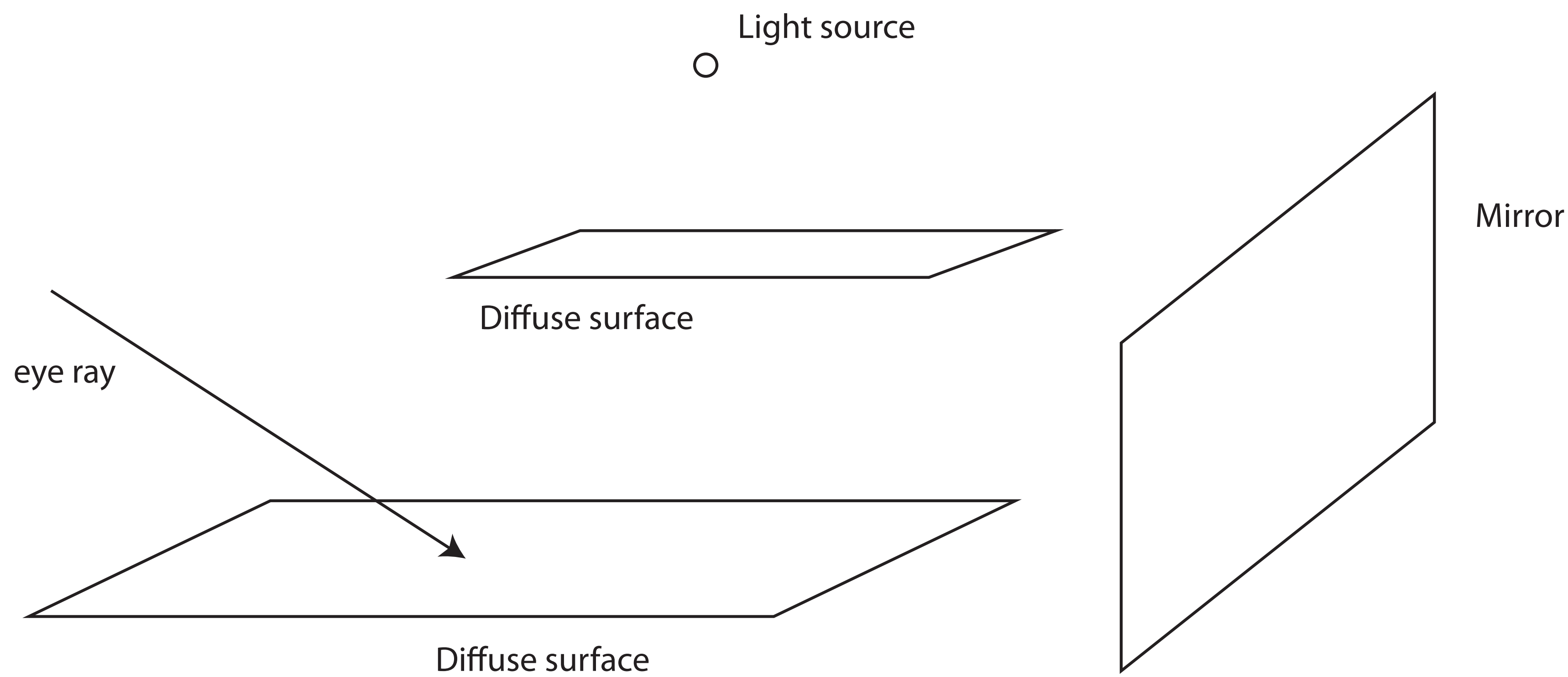
LDD+E - easy geometry, but variance



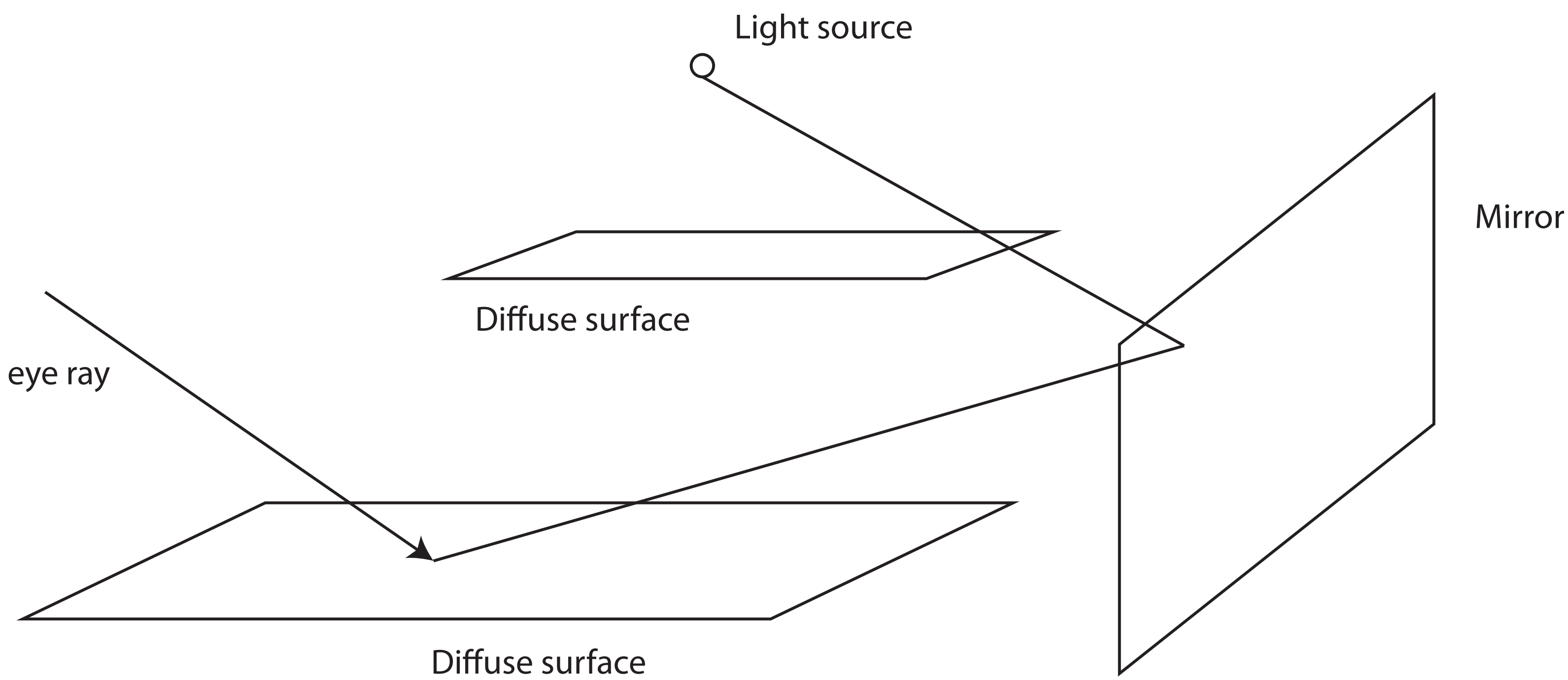
LDD+E - variance control (sketch!)

- In principle, easily sampled recursively
- Preferentially sample paths that make large contributions
  - these are paths that connect light, eye, via high albedo surfaces
    - “Russian roulette”
      - continue path with probability = albedo
      - weight by (1/albedo)
- OR Cache results
  - propagating a path:
    - check: is there something in the cache?
      - yes: use it
      - no: propagate path and cache results

LS+DE - harder - where is the light?



LS+DE - harder - where is the light?



Problem: which direction leaving diffuse surface will hit the light?

Strategies

- Bidirectional ray tracing
  - Trace a lot of rays from light through specular surfaces to first diffuse
  - Trace a lot of eye rays to first diffuse
  - Join paths
  - Variance control
    - weight paths as if they'd been found in different ways (Veach +Guibas)
- Markov chain monte carlo
  - take a path and mutate it; reweight contribution of mutated path
- Caching
  - Photon cache - trace many rays from light through specular to diffuse
    - cache at diffuse
    - query with eye ray

Biased vs unbiased rendering

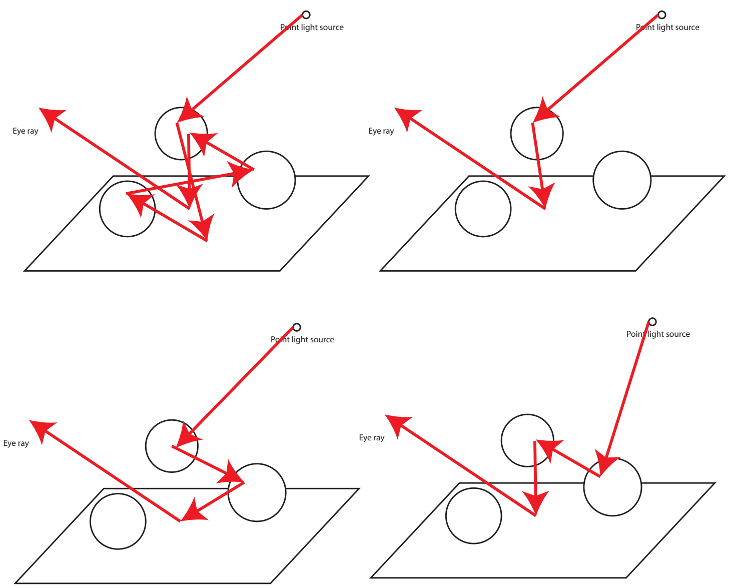
- Unbiased renderer
  - pixel value is value of random variable (different paths=different values)
  - E(estimate)=True value
  - sometimes essential
    - eg estimate the amount of light in a museaum hall
- Biased renderer
  - E(estimate)=True value + Bias
  - eg Photon map (as above)
    - Bias because we do not know how many photons to stick in cache
  - Often more realistic
- Crucial point
  - Very few people can tell if a render is nearly physically correct
    - and no-one can reliably spot exactness

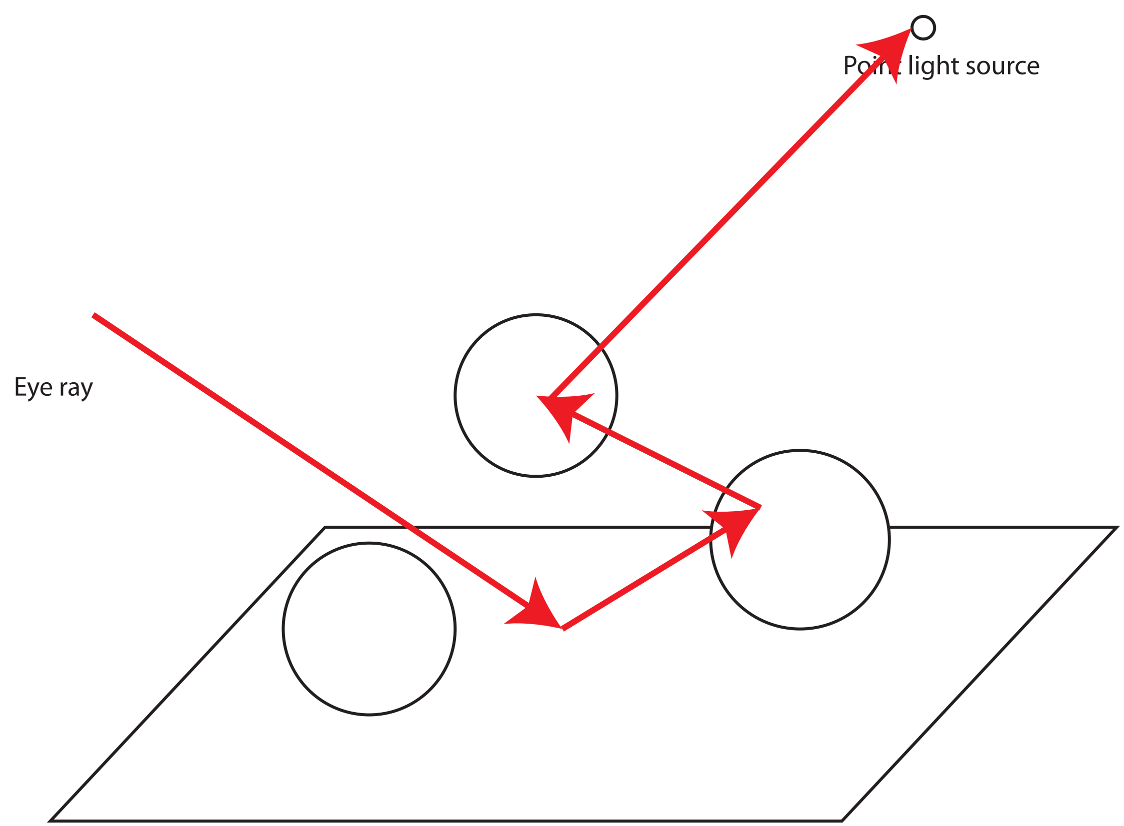
The plenoptic function

- We are repeatedly sampling radiance
  - as function of point, direction
- What if we had a function that could report that?

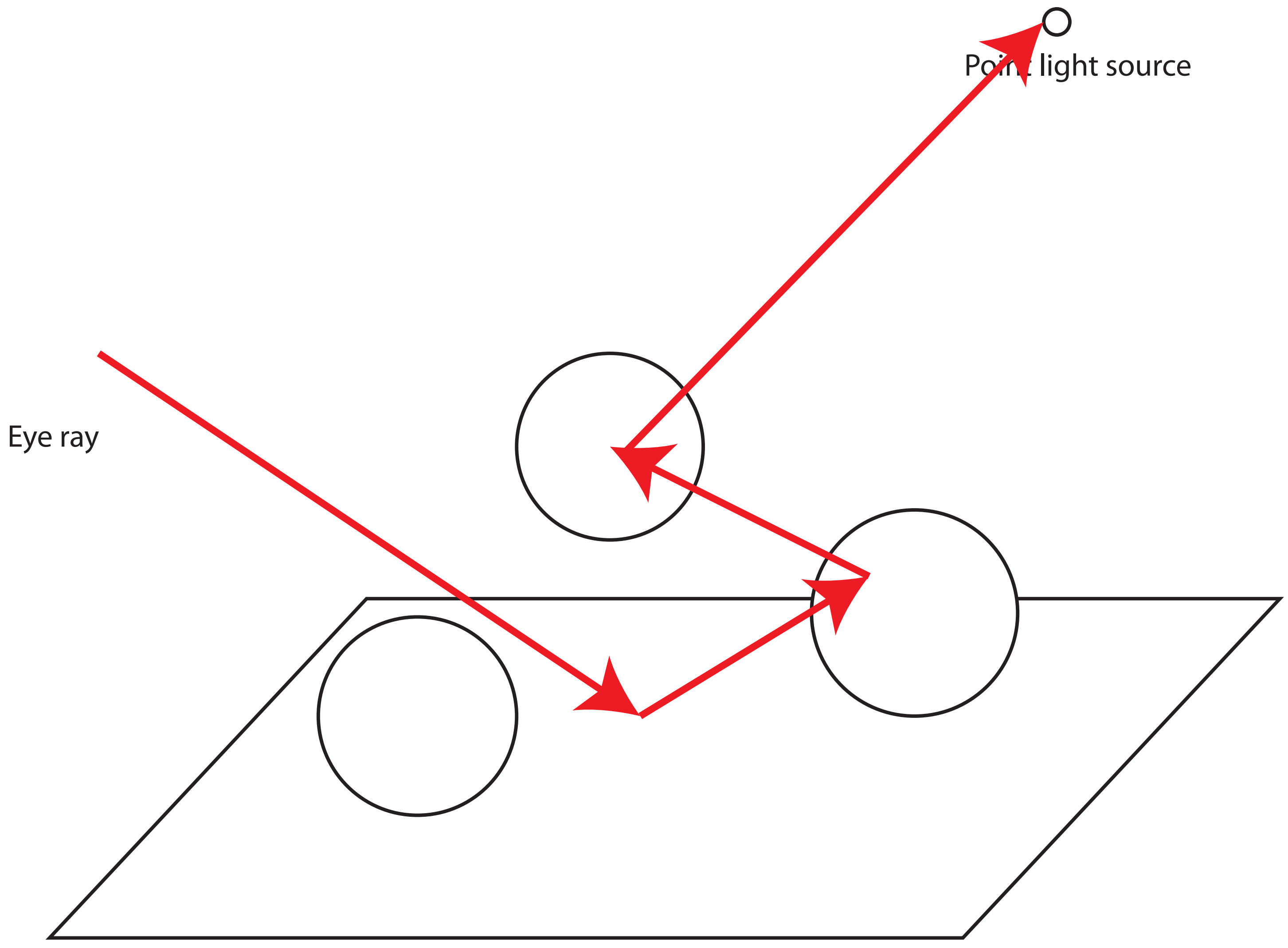
- Plenoptic function
  - radiance along a directed line
  - space of lines is rather nastier than you might think

The plenoptic function as a cache





The plenoptic function as a cache



The plenoptic function - careful!

- This is a function whose domain is nasty
  - all maximal directed line segments (lines) in free space
  - the domain can get very complicated
    - easy when there aren't any objects
    - otherwise, much harder
  - domain is sometimes called a visibility complex

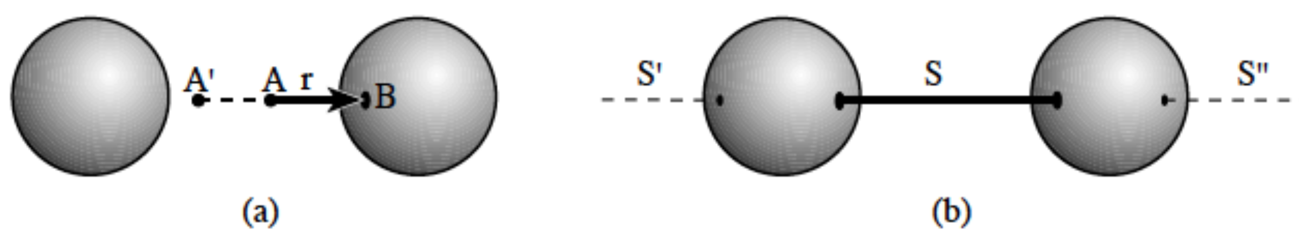
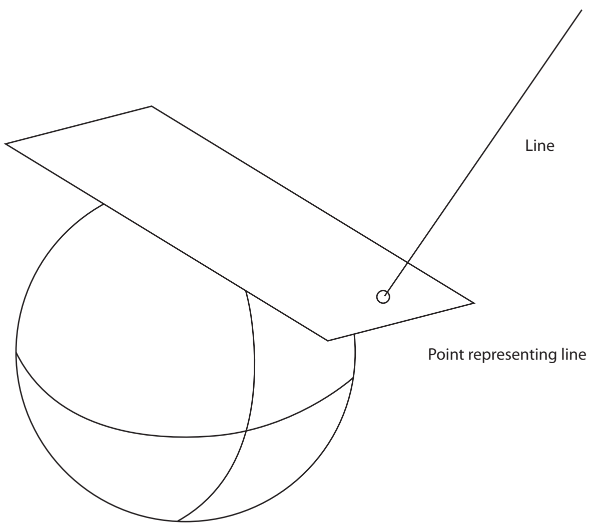


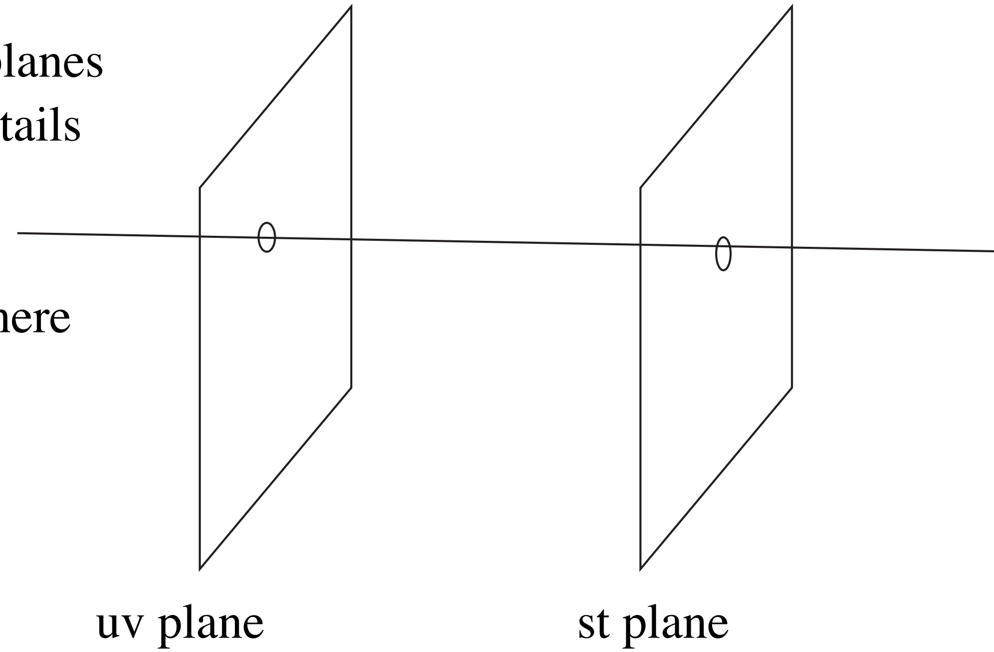
Fig. 1. Maximal free segment. (a) All the rays collinear to r whose origin is between the two spheres "see" point B. (b) These rays are grouped into a *maximal free segment* S. Two other maximal free segments S' and S'' are collinear to S.

Durand et al 02

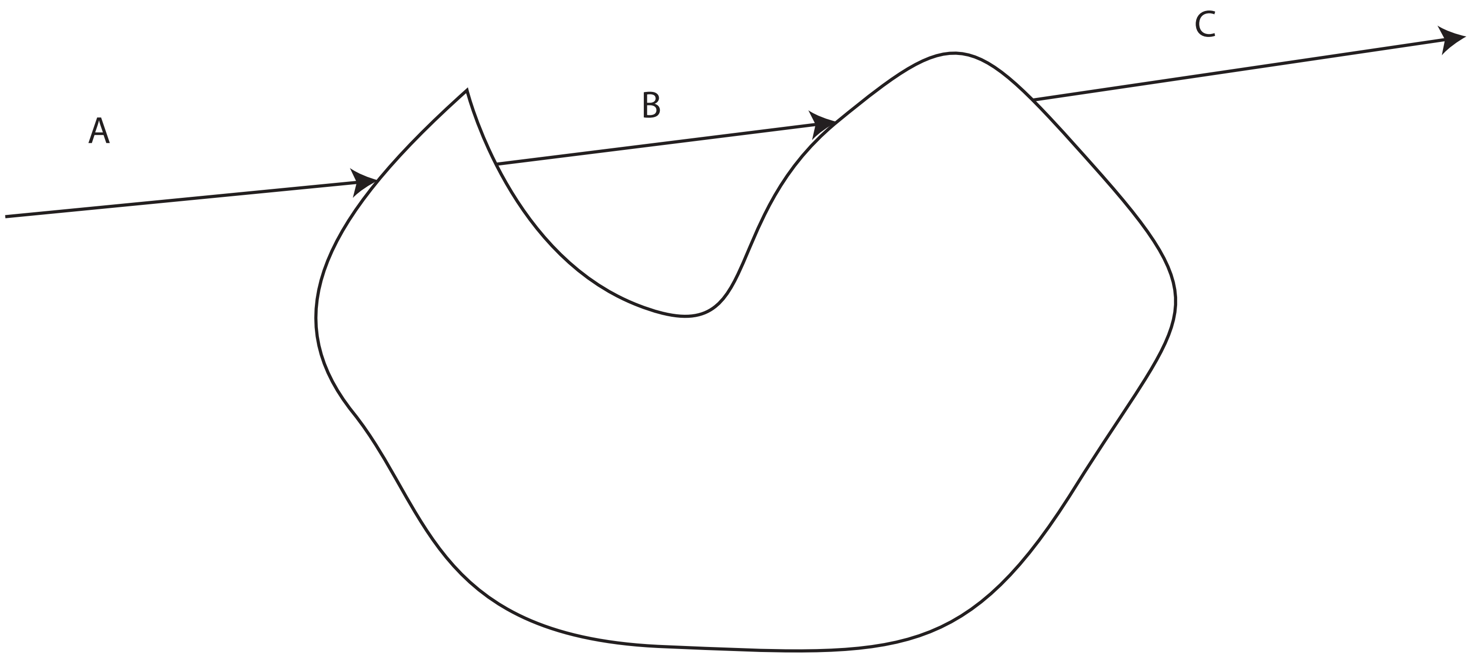
Lines in 3D (if it's empty!)

- Space of lines is 4 dimensional
  - can specify a line by:
    - where it intersects each of two planes
      - some missing lines, some details
  - alternative
    - directed line
    - point on the tangent plane of sphere





Lines in 3D with object can be nasty



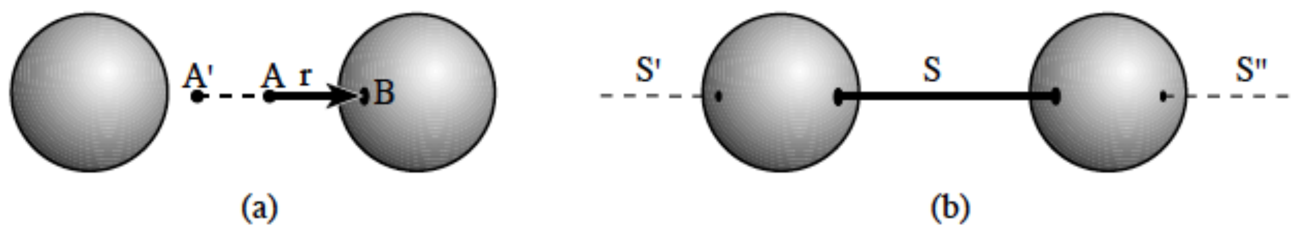
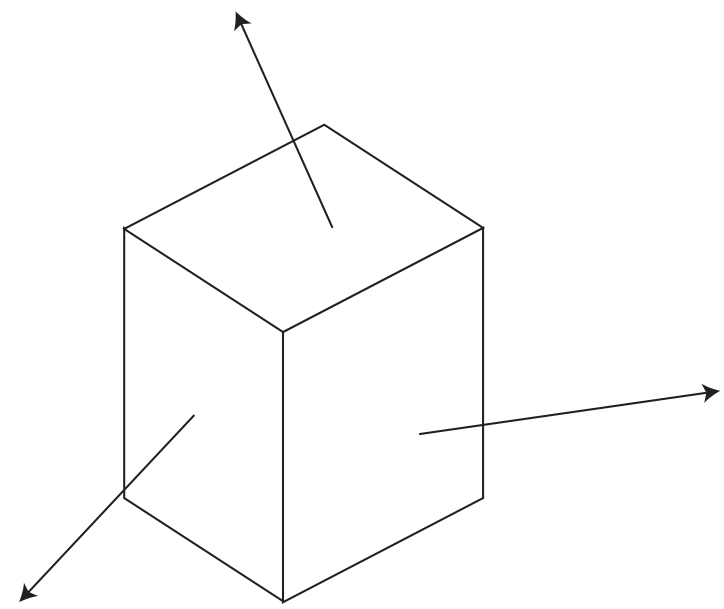


Fig. 1. Maximal free segment. (a) All the rays collinear to r whose origin is between the two spheres "see" point B. (b) These rays are grouped into a *maximal free segment* S. Two other maximal free segments S' and S'' are collinear to S.

Durand et al 02

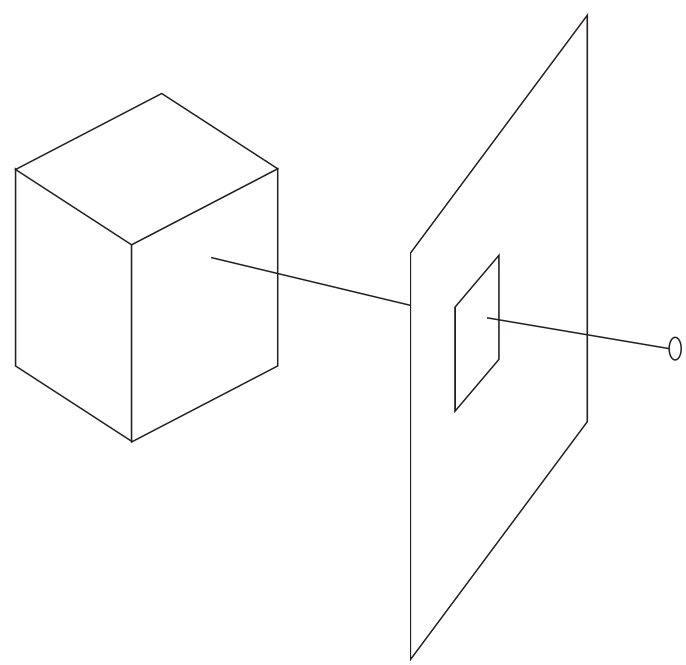
Simplify

- Place an object in a box
  - record radiance for each ray leaving box
- Easy to ray trace
  - look up eye ray in rays leaving box
    - report that value
- Capture is relatively easy



Capturing this representation

- Obtain an awful lot of images from calibrated cameras
  - each image is a set of rays leaving the box
  - calibration



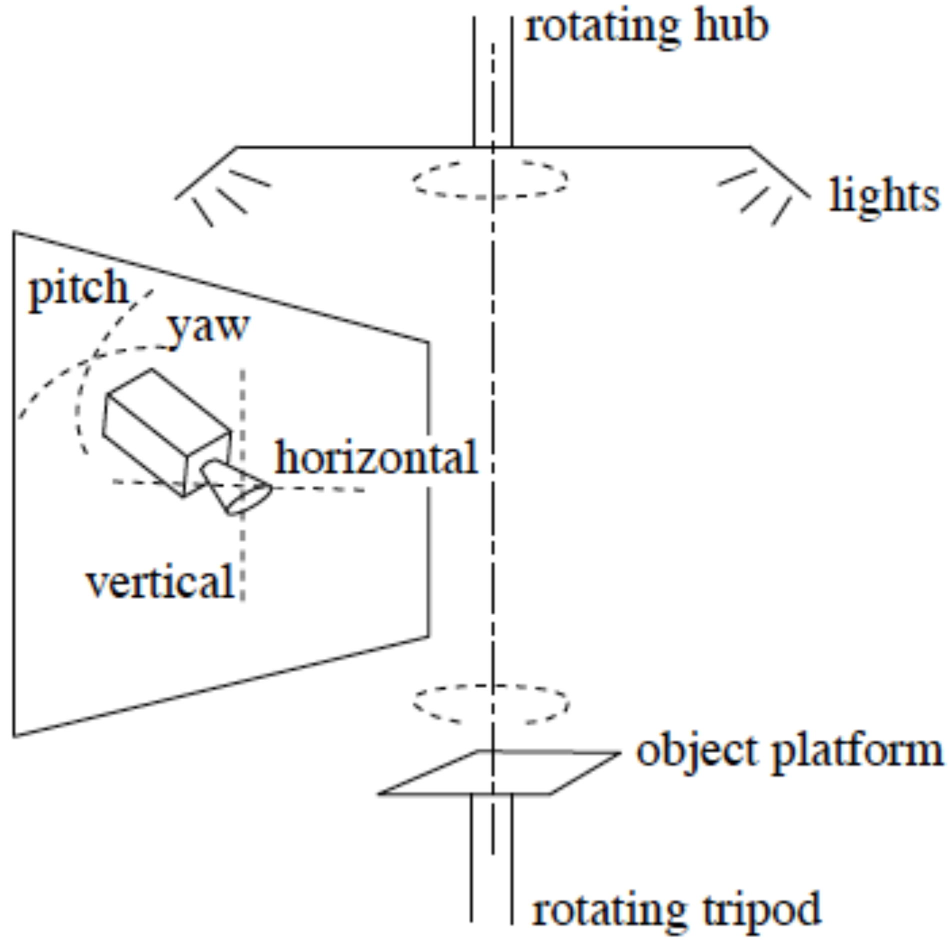


Figure 10: Object and lighting support. Objects are mounted on a Bogen fluid-head tripod, which we manually rotate to four orientations spaced 90 degrees apart. Illumination is provided by two 600W Lowell Omni spotlights attached to a ceiling-mounted rotating hub that is aligned with the rotation axis of the tripod. A stationary 6' **x** 6' diffuser panel is hung between the spotlights and the gantry, and the entire apparatus is enclosed in black velvet to eliminate stray light.

Levoy and Hanrahan, 96

Rendering

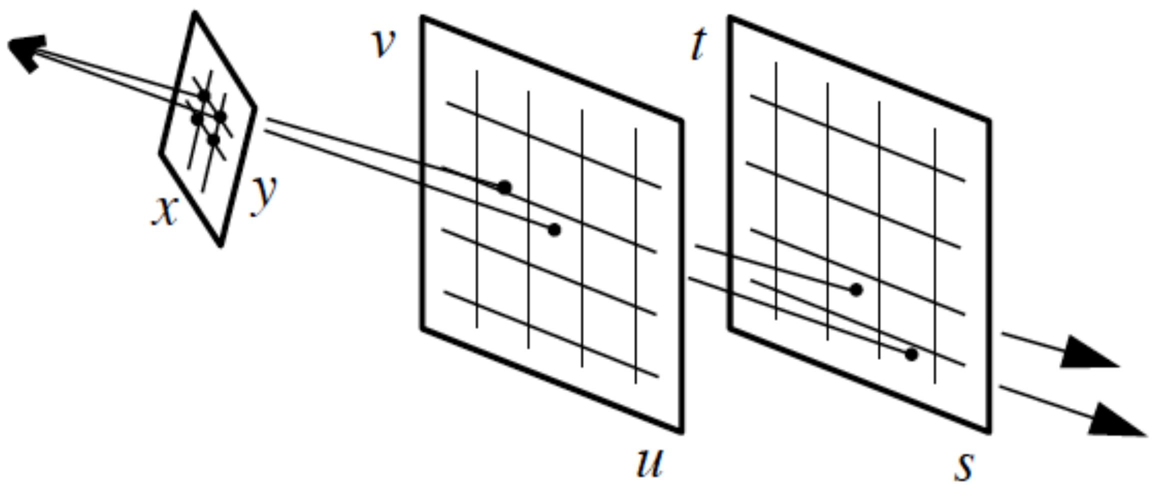


Figure 12: The process of resampling a light slab during display.

Levoy+Hanrahan, 96

Issue: Sampling and Interpolation

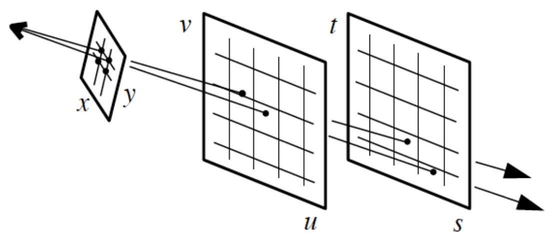


Figure 12: The process of resampling a light slab during display.

- Almost every eye ray ends up "between" uv, st samples
  - we must interpolate (smooth; something)
  - Traditional: multilinear interpolation

Interpolation helps, but..

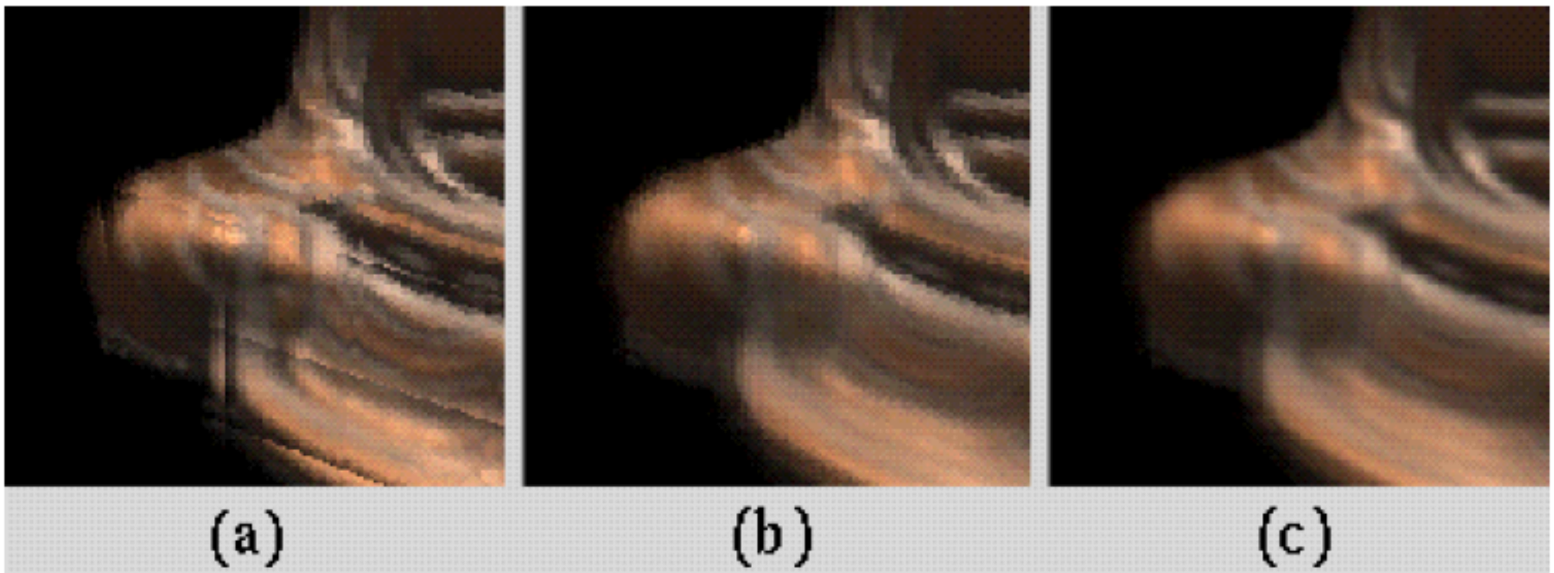
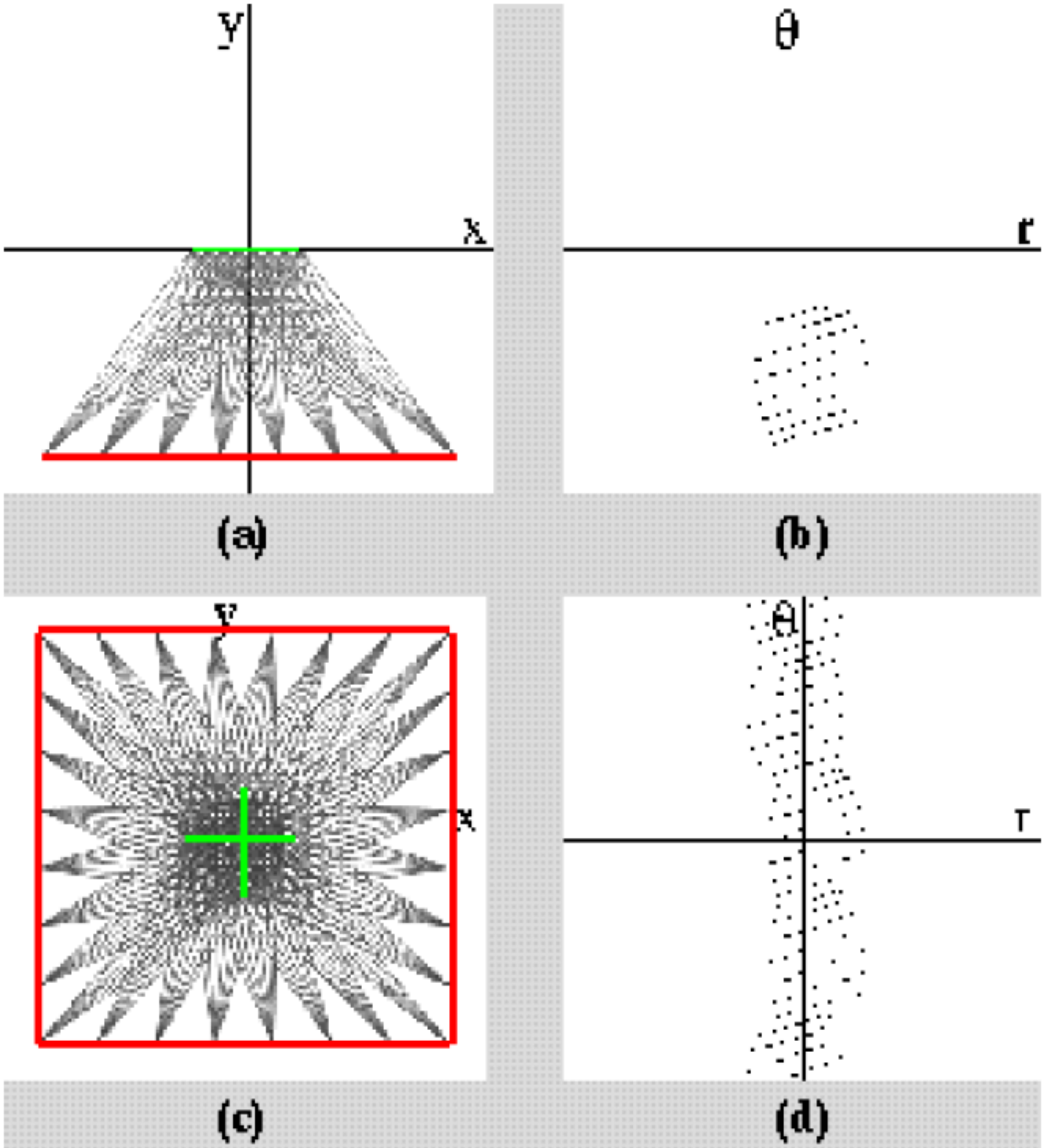


Figure 13: The effects of interpolation during slice extraction. (a) No interpolation. (b) Linear interpolation in uv only. (c) Quadra-linear interpolation in uvst.



Two plane representation and sampling

Levoy+Hanrahan, 96

Figure 3: Using line space to visualize ray coverage. (a) shows a single

Revise model

- We need:
  - better interpolation
  - easier capture
  - some way to deal with the awkwardness of line representations
- Ideas:
  - move to scattering/volume rendering based representation
    - this will make the line representation easier to deal with
  - use a multilayer perceptron to represent relevant functions

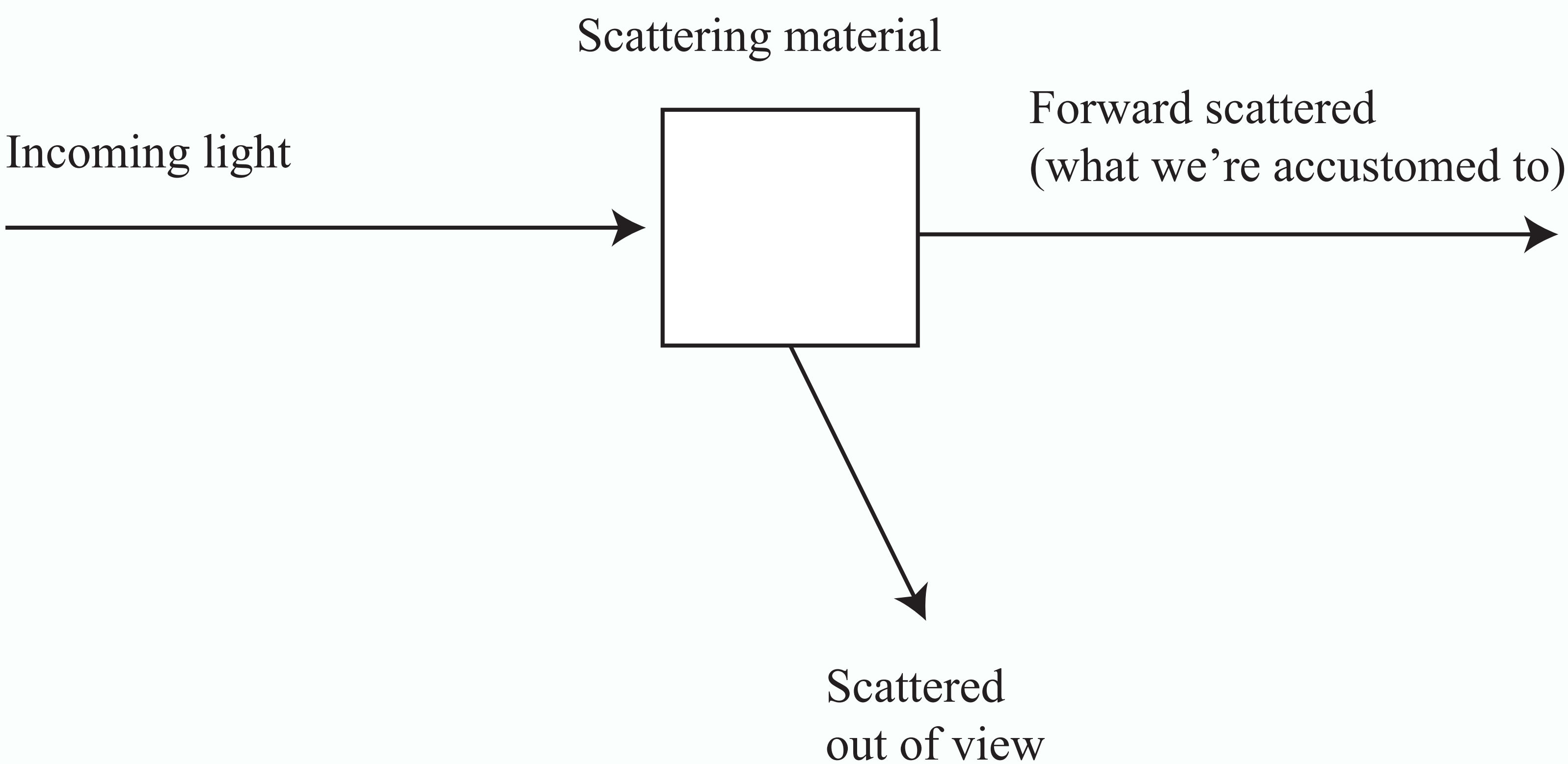
Scattering

- Fundamental mechanism of light/matter interactions
- Visually important for
  - slightly translucent materials (skin, milk, marble, etc.)
  - participating media
- In fact, it's the mechanism underlying reflection

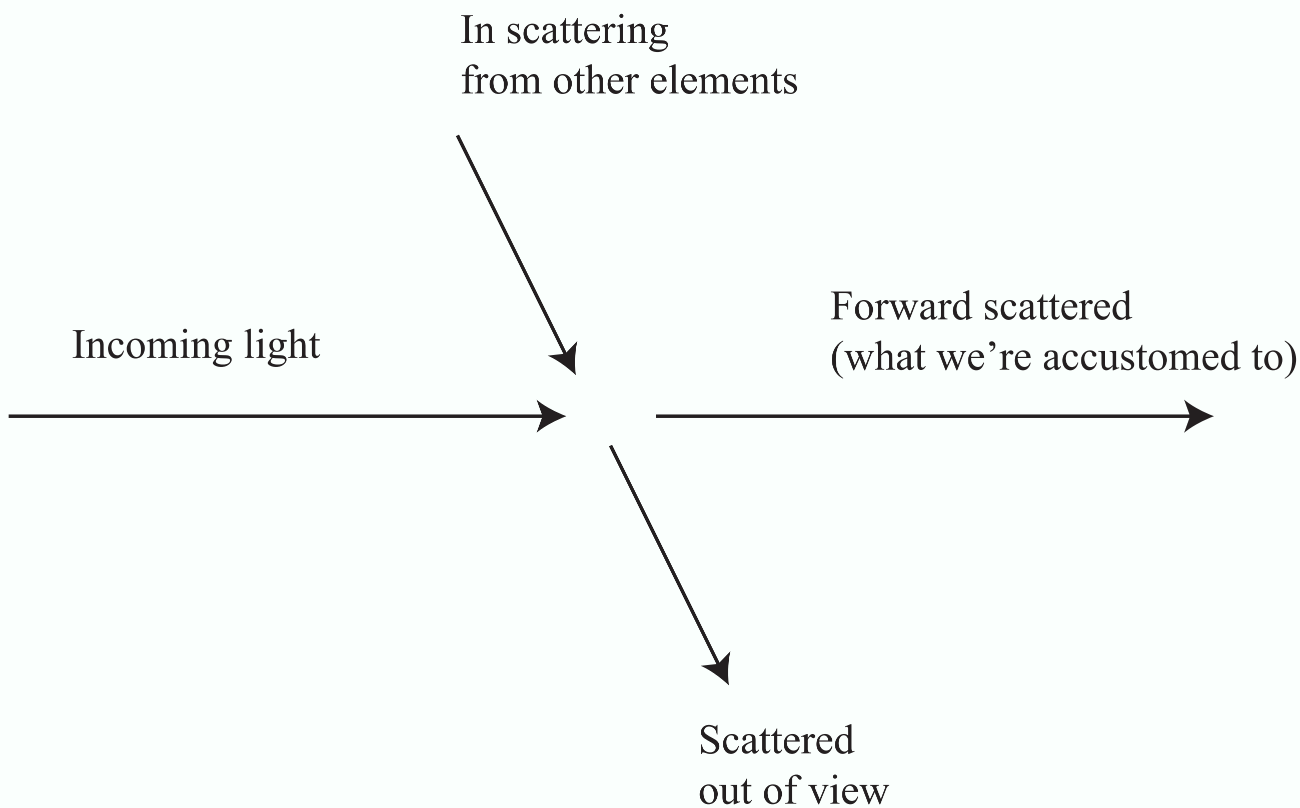
Participating media

- for example,
  - smoke,
  - wet air (mist, fog)
  - dusty air
  - air at long scales
- Light leaves/enters a ray travelling through space
  - leaves because it is scattered out
  - enters because it is scattered in
- New visual effects

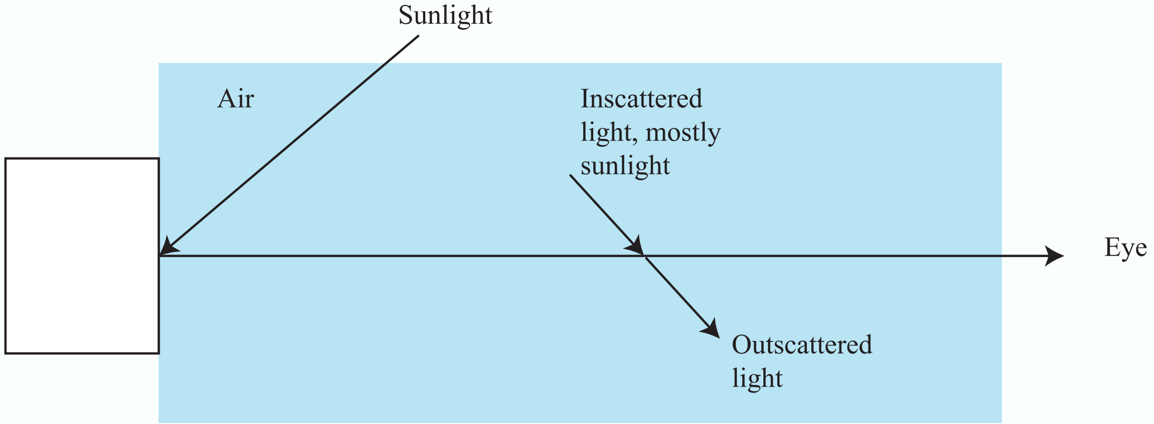
Light hits a small box of material



A ray passing through scattering material

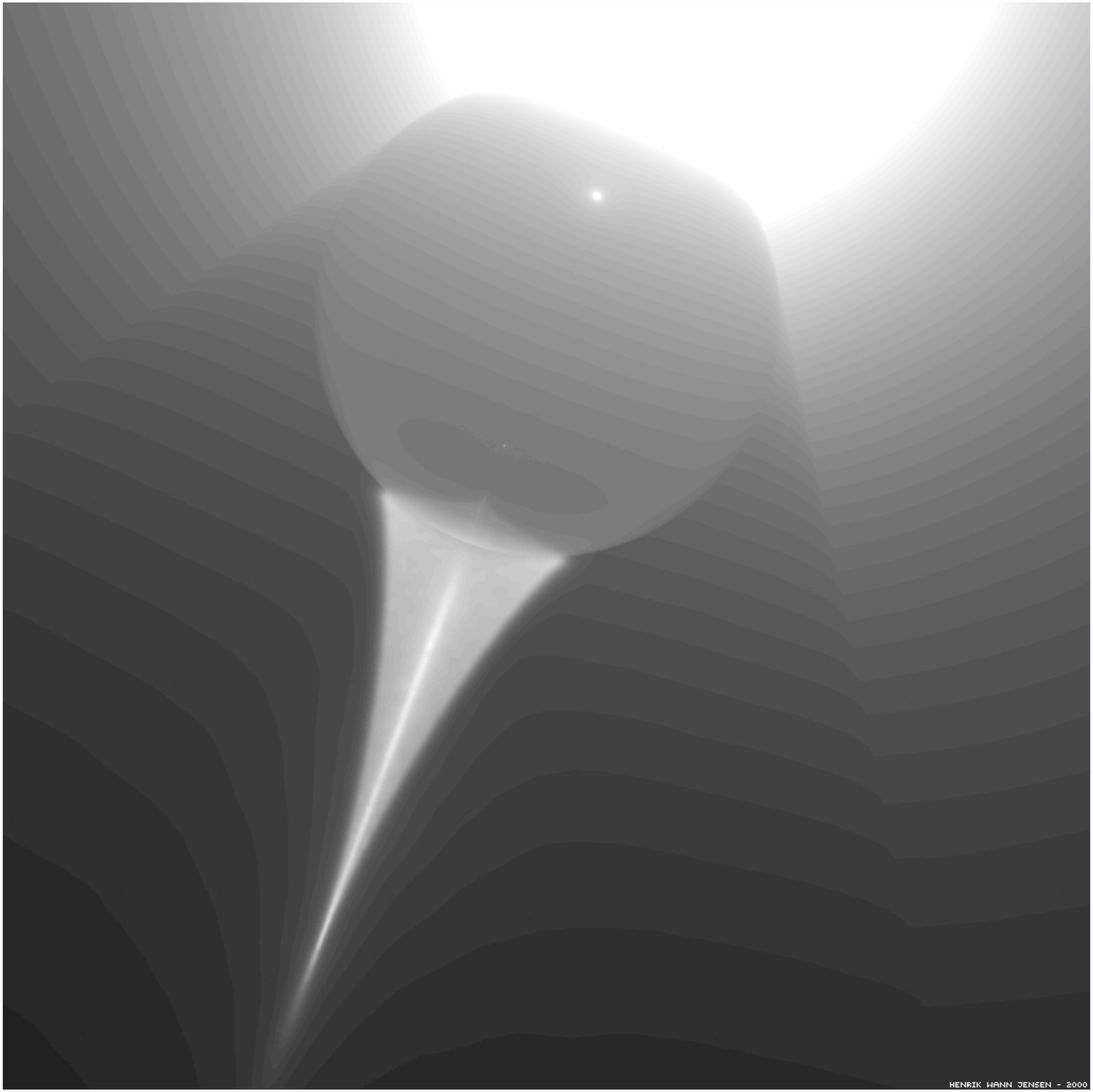


Airlight as a scattering effect



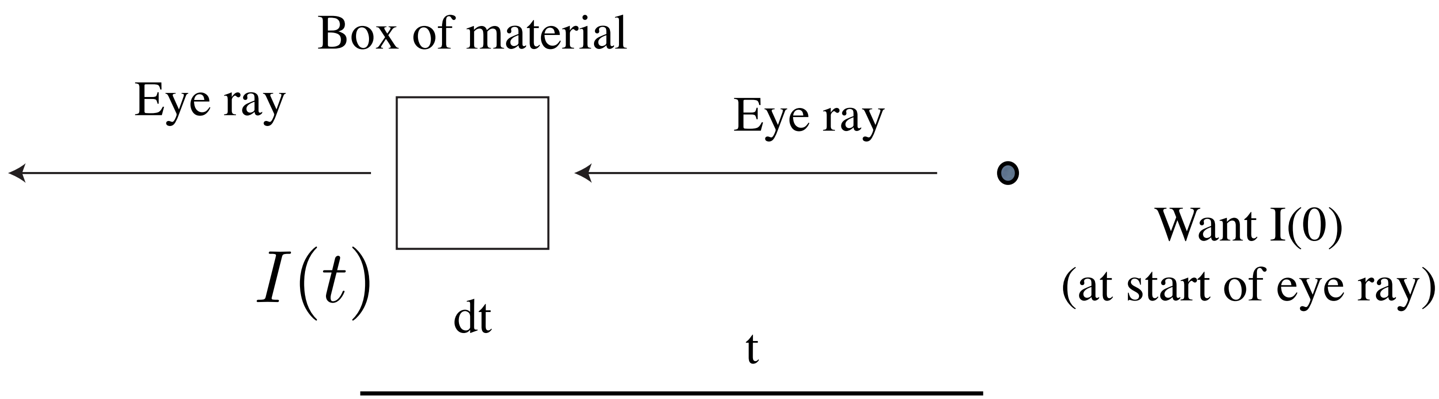

original unique filename: 20180329-141700_baie_des_fourmis.jpg
hosted by www.carto.net
photo ©André M. Winter

From Lynch and Livingstone, Color and Light in Nature

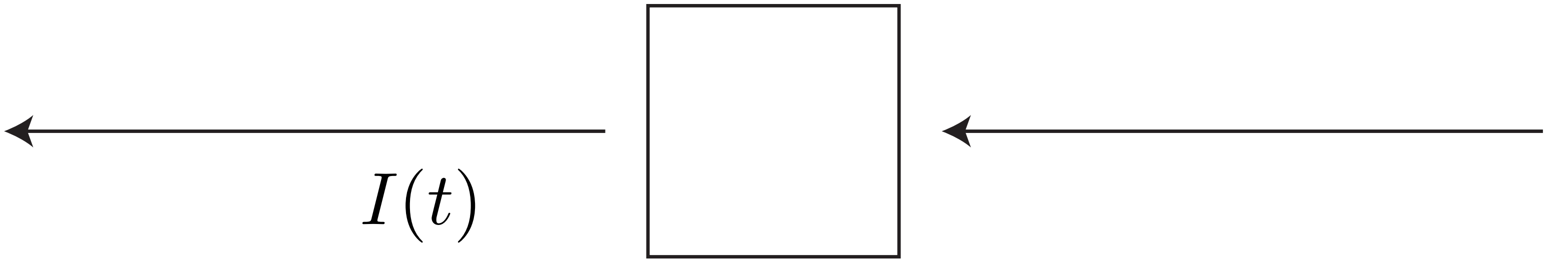

HENRIK WANN JENSEN - 2000

From Lynch and Livingstone, Color and Light in Nature

Rendering this



- Ignore in-scattering
  - only account for forward scattering
- Assume there is a source at t=T
  - of intensity I(T)
  - what do we see at t=0?



$$
\begin{aligned}
I(t-\delta t) &= I(t) - \delta I \\
&= I(t) - \sigma(t) I(t)
\end{aligned}
$$

$$\frac{dI}{dt} = \sigma(t) I(t)$$

$$\frac{d \log I}{dt} = \sigma(t)$$

$$I(T) = I(0) e^{\int_0^T \sigma(t) dt} \qquad I(0) = I(T) e^{-\int_0^T \sigma(t) dt}$$

Yields

The volume density $\sigma(\mathbf{x})$ can be interpreted as the differential probability of a ray terminating at an infinitesimal particle at location $\mathbf{x}$. The expected color $C(\mathbf{r})$ of camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ with near and far bounds t_n and t_f is:

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt\,, \text{ where } T(t) = \exp\left(-\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds\right). \quad (1)$$

Mildenhall et al, 20