Lecture 19: December 1, 1998

Today:

Radiosity (continuation of Global Illumination)

HKN reviews

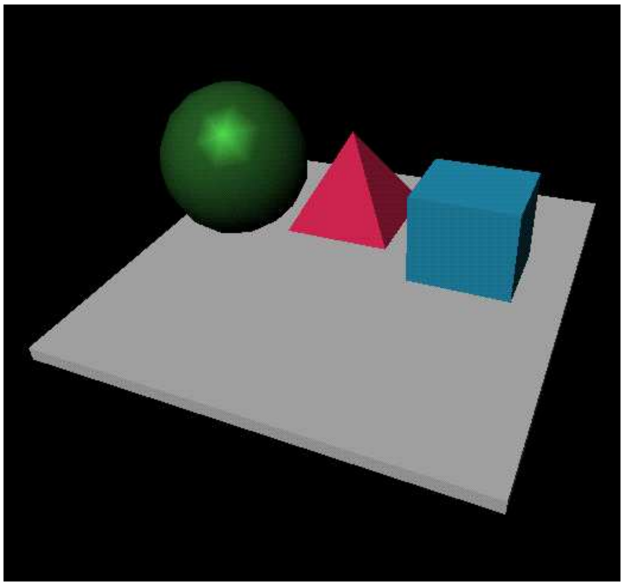
Thursday:

Texture Mapping

Scan Conversion and Shading

Shading computed with a *constant* amount of state

Typically, some number of h/w light sources



Consequences:

plastic look; incorrect highlights; no

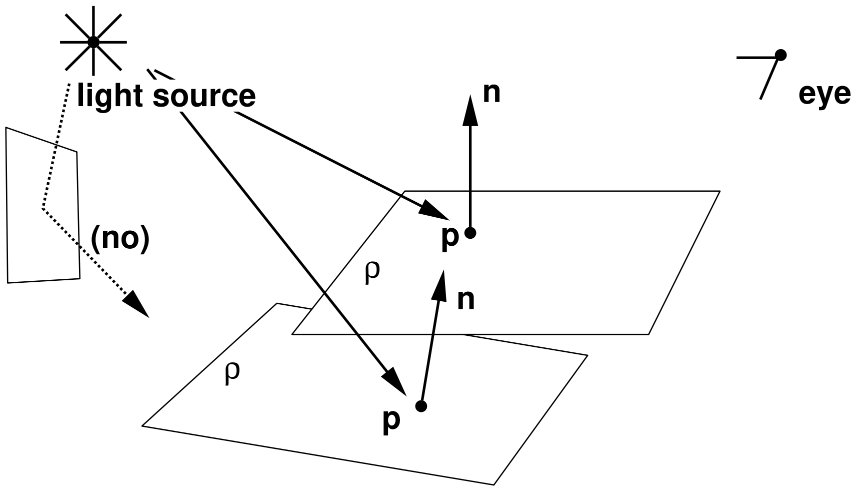
shadows

secondary illumination

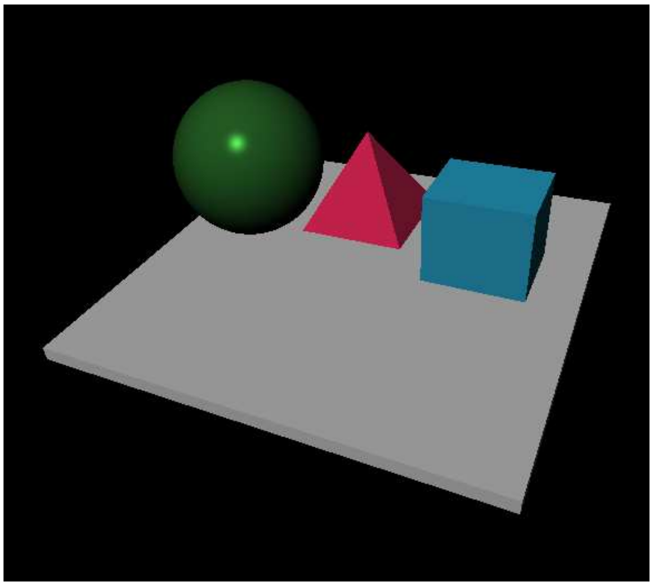
transmission

focusing effects

Hardware Shading



Point light sources only
No occlusion testing
Primary light sources only



```
% ivray raycast.iv -e raycast.env -s L
```

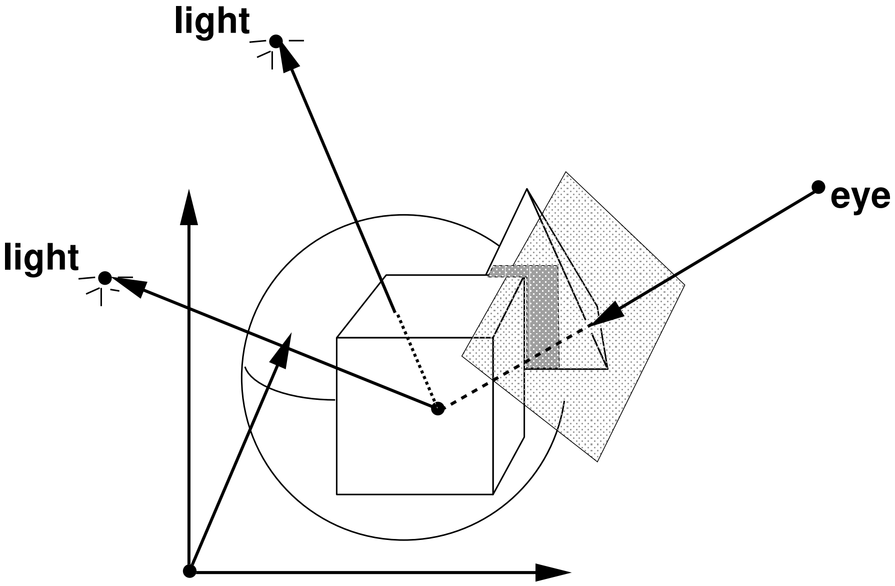
Alternative: Ray Casting, Semi-Local Shading

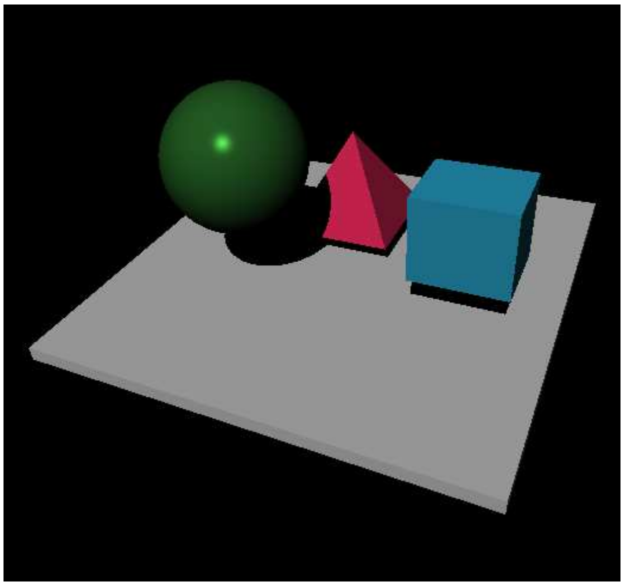
Idea (Appel, 1968):

- Cast ray from eye through each pixel
- Determine closest object along ray
- Shade by summing *unoccluded* lights

Non-recursive! (But improved quality, realism)

(Primary) shadows easily handled





```
% ivray raycast.iv -e raycast.env -s S
```

Recursive Ray Tracing

Extend to reflection, refraction

Clearly shading must take entire scene into account

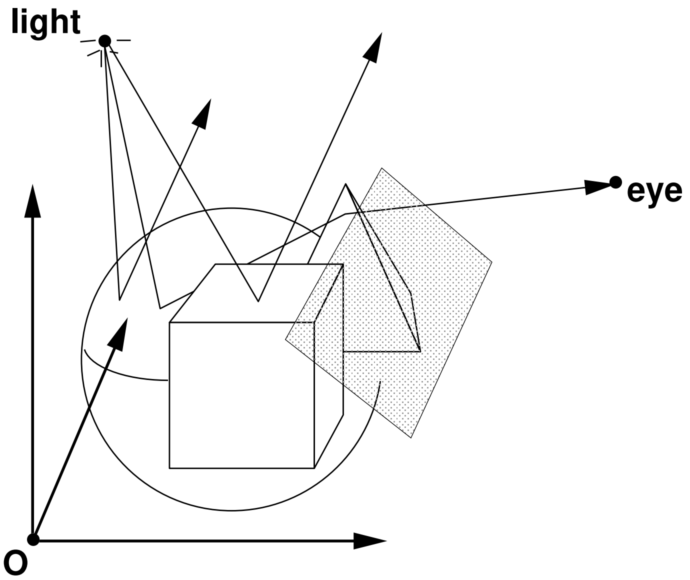
Ray tracing is a "global illumination algorithm"

Idea: Light *originates* at light sources, so

"trace" photon paths from the light source

Known as **Forward Ray Tracing:**

At each interaction, surface properties dictate absorption, reemission, transmission probability



Disadvantages?

Forward Ray Tracing

Disadvantages

very few of the photons end up at the eye

very hard to know in which directions
photons should be sent

enormous number of cycles expended per photon
(can be ameliorated by *packet tracing*)

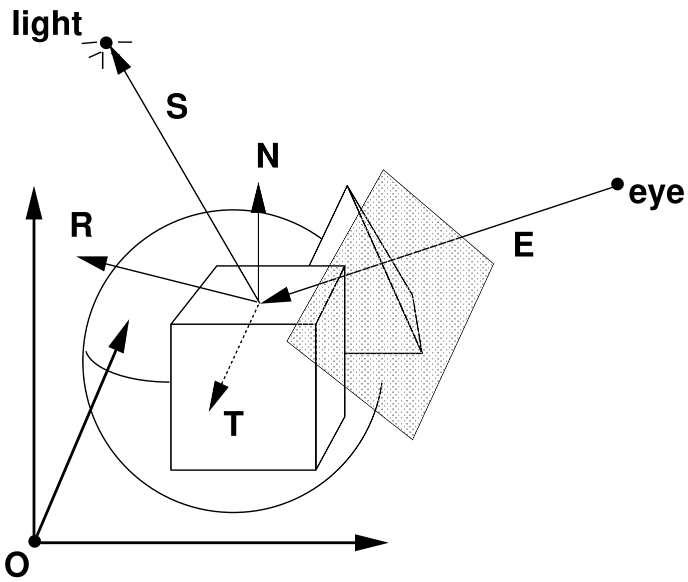
result is usually objectionable *noise*

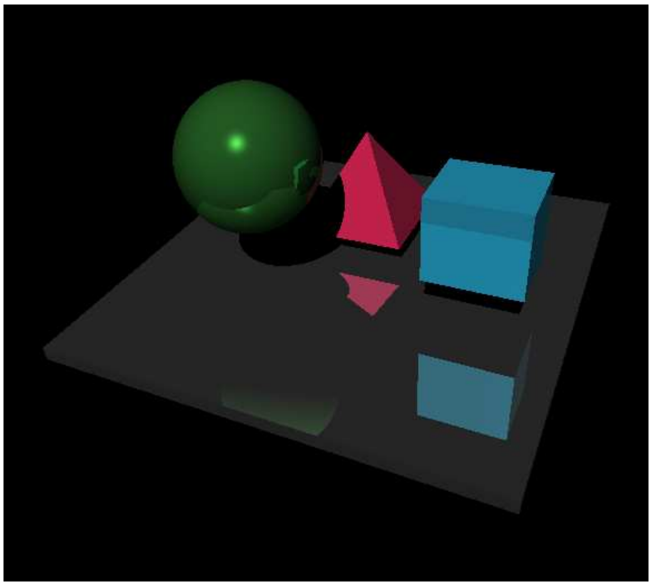
Backward Ray Tracing

Insight: we only "see" rays that make it to eye

So, trace "eye rays" **E** *backward* into scene

Find contributions to shading at surface points





```
% ivray raycast.iv -e raycast.env -s R -d 1
```

Shadow rays **S** (to light sources)

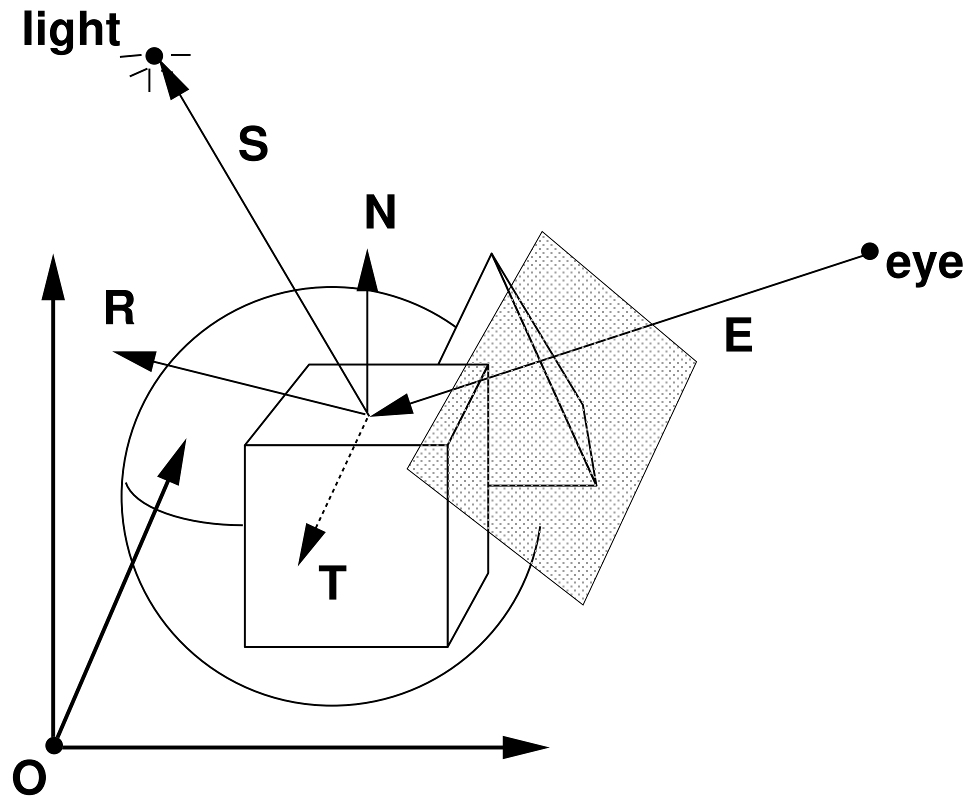
Reflection rays **R** (along specular direction)

Refraction rays **T** (along refraction direction)

Note: Shading operation is **recursive** !

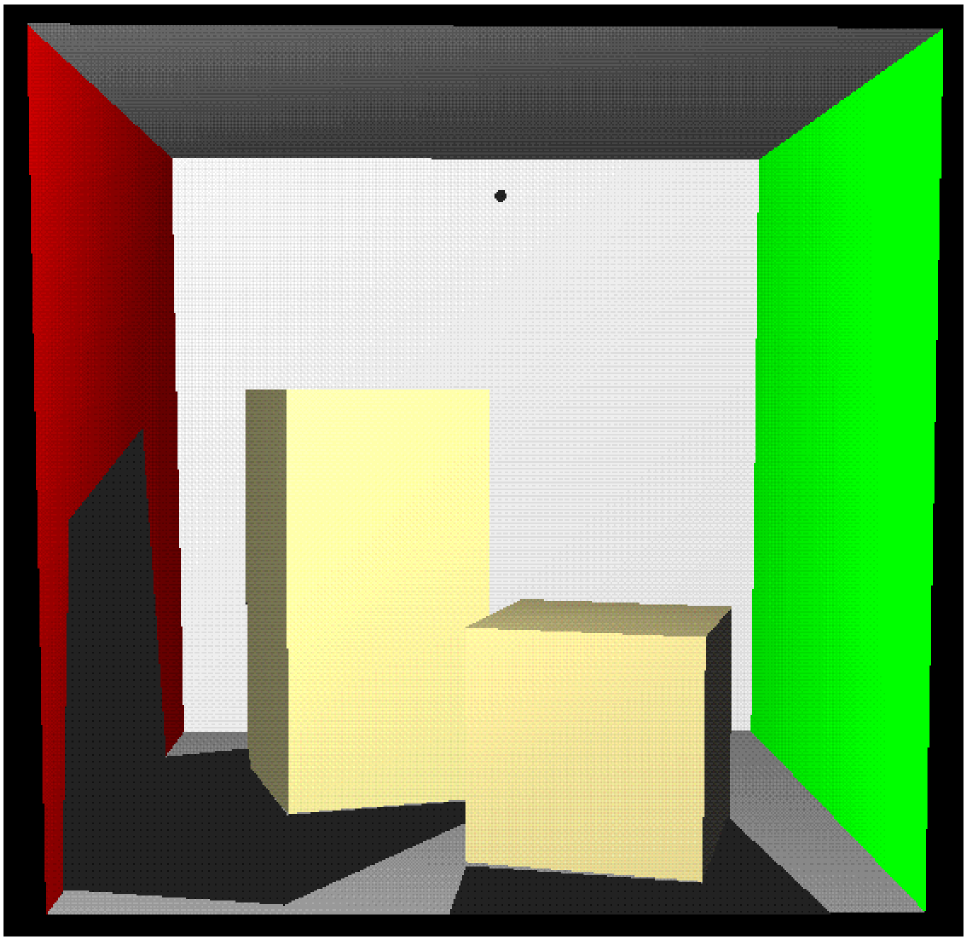
Radiosity Solvers

One possibility: modify ray-tracer to sample area light sources, hemispherical irradiance:



this requires lots of CPU for ray casting
any other disadvantages ?

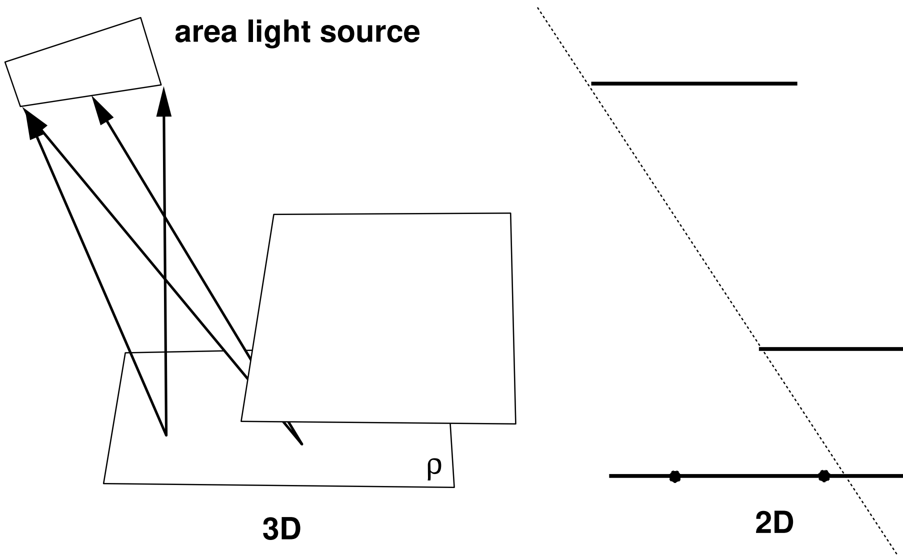
Hard shadows



Abrupt changes in illumination arise in ray tracing

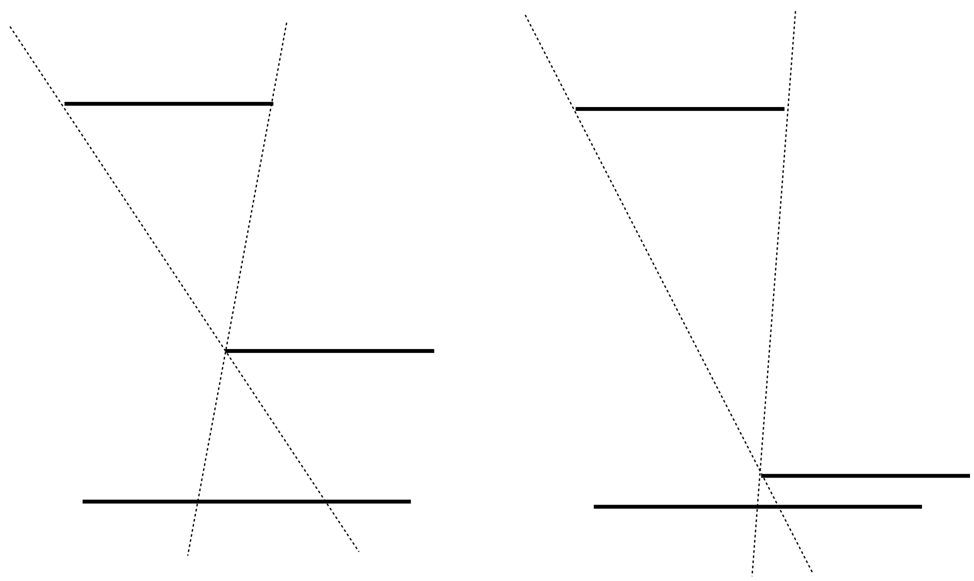
"Point" light source is non-physical; it goes abruptly from visible to invisible

Soft shadows



Illumination Gradients

What if occluder is relatively far from receiver?
What if occluder is relatively close to receiver?



Raytracing

Typically specular materials; small / distant lights
Traces light paths terminating at the eye
Produces view–dependent solution: set of pixels

Courtesy Don Mitchell, AT&T Bell Labs

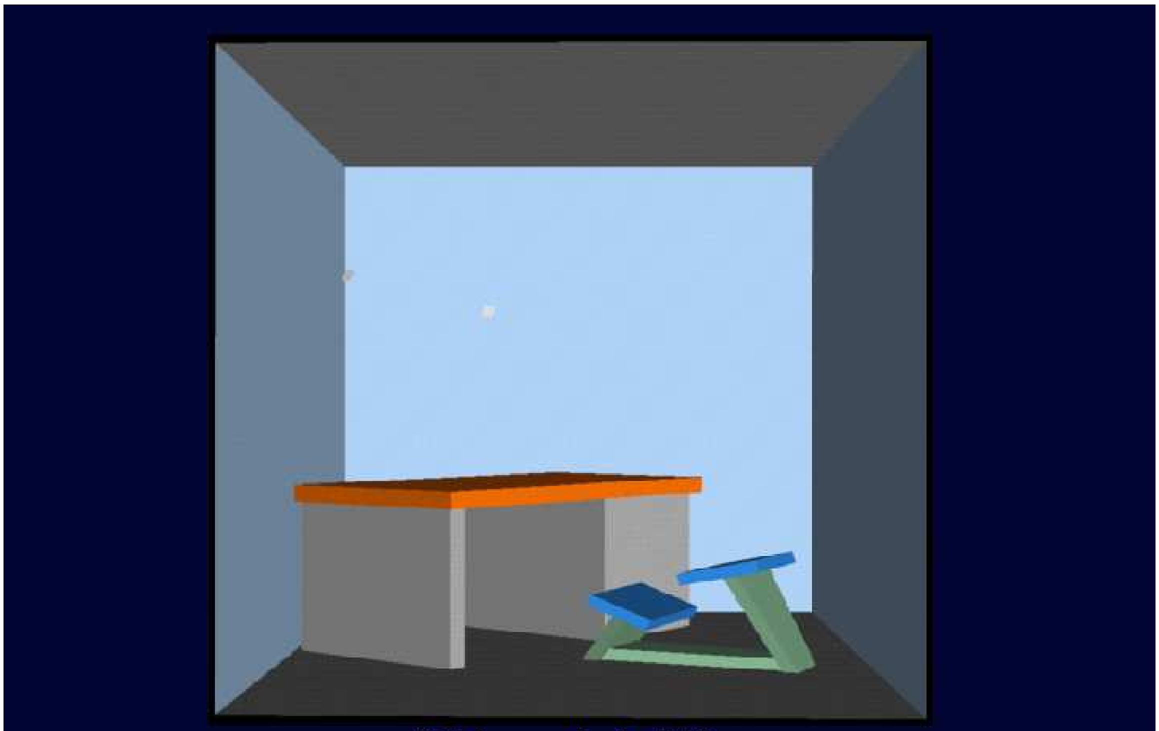
Radiosity

- Typically diffuse materials; large / local lights
- Traces all light paths bearing significant energy
- Produces view–independent surface illumination

Radiosity Methods

Arose from thermal radiation literature
(Analyzing heat thrown from blast furnaces)
Find equilibrium distribution of radiant energy
Suitable for ideal *diffuse* environments

Radiosity Equation - Inputs and Outputs

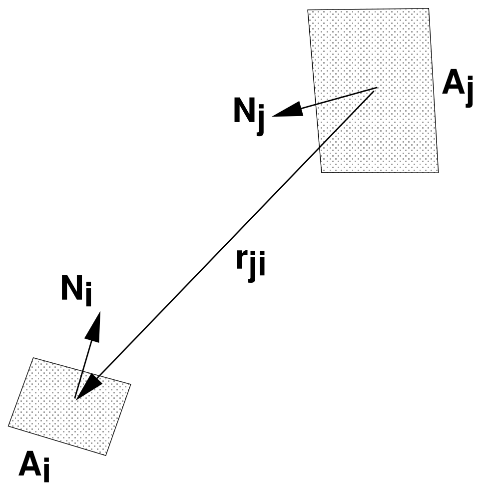

Patches and reflectivities

Radiosity Equation - Intuitive Form

Radiosity equation (B,E units power per area):

$$\begin{aligned}
\text{Exitant Power} &= \text{Emitted Power} + \text{Reflected Power} \\
B_i A_i &= E_i A_i + \rho_i (\text{Total Incident Power}) \\
&= E_i A_i + \rho_i \sum_j (\text{Incident Power from } j) \\
&= E_i A_i + \rho_i \sum_j B_j A_j F_{ji}
\end{aligned}$$

Where F_{ji} is the **fraction of radiant power leaving j that arrives at i**



(Or, fraction of j's hemisphere occupied by i, integrated over all points on j)

F_{ji} called the "Form Factor" between j and i

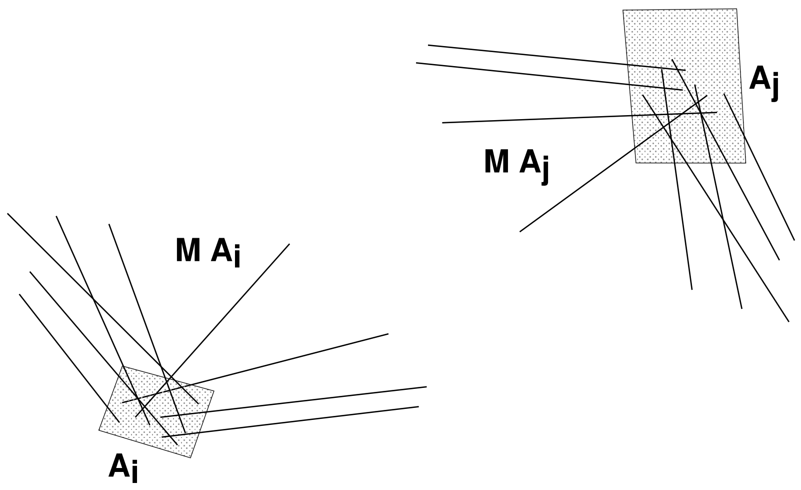
Note order, interpretation of subscripts!

Radiosity Equation: Computational Form

Geometric fact:

$$A_j F_{ji} = A_i F_{ij}$$

Proof: lines can be counted under some measure



\# of lines through A_i is MA_i
\# of lines through A_j is MA_j
Why? What is M?
So # of lines leaving A_i arriving at A_j is MA_iF_{ij},
and # of lines leaving A_j arriving at A_i is MA_jF_{ji}
But both quantities describe the same set of lines!

Radiosity Equation: Computational Form

Relate total power as follows

$$\begin{aligned} B_i A_i &= E_i A_i + \rho_i \sum_j B_j A_j F_{ji} \\ &= E_i A_i + \rho_i \sum_j B_j F_{ij} A_i \end{aligned}$$

This is the "power equation"

Dividing power equation through by A_i yields

$$B_i = E_i + \rho_i \sum_j B_j F_{ij}$$

This is the "radiosity equation"

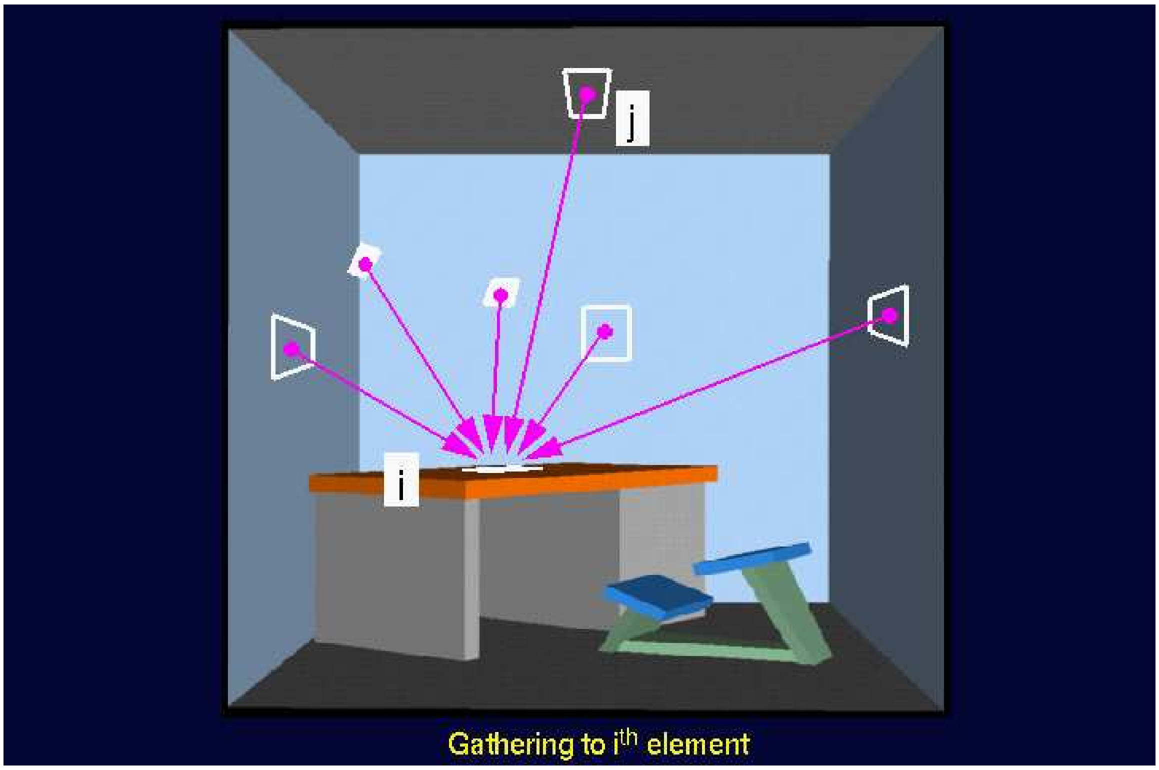
More efficient, but more confusing

Be careful to get subscripts right!

Solving the Radiosity Equation

Can't be done in closed form (in general)
  so use *numerical methods* (what are they?)
Set up initial conditions (for all i):
  Assign E_i:
    $E_i > 0$ (light sources)
    $E_i = 0$ (all other surfaces)
  Assign B_i, initial guess for i^{th} surface's radiosity
    What is a good heuristic for initial guess?
  Determine **geometric** coupling:
    Compute F_{ij} for all i, j
    (How long does this take?)

Gather Operation

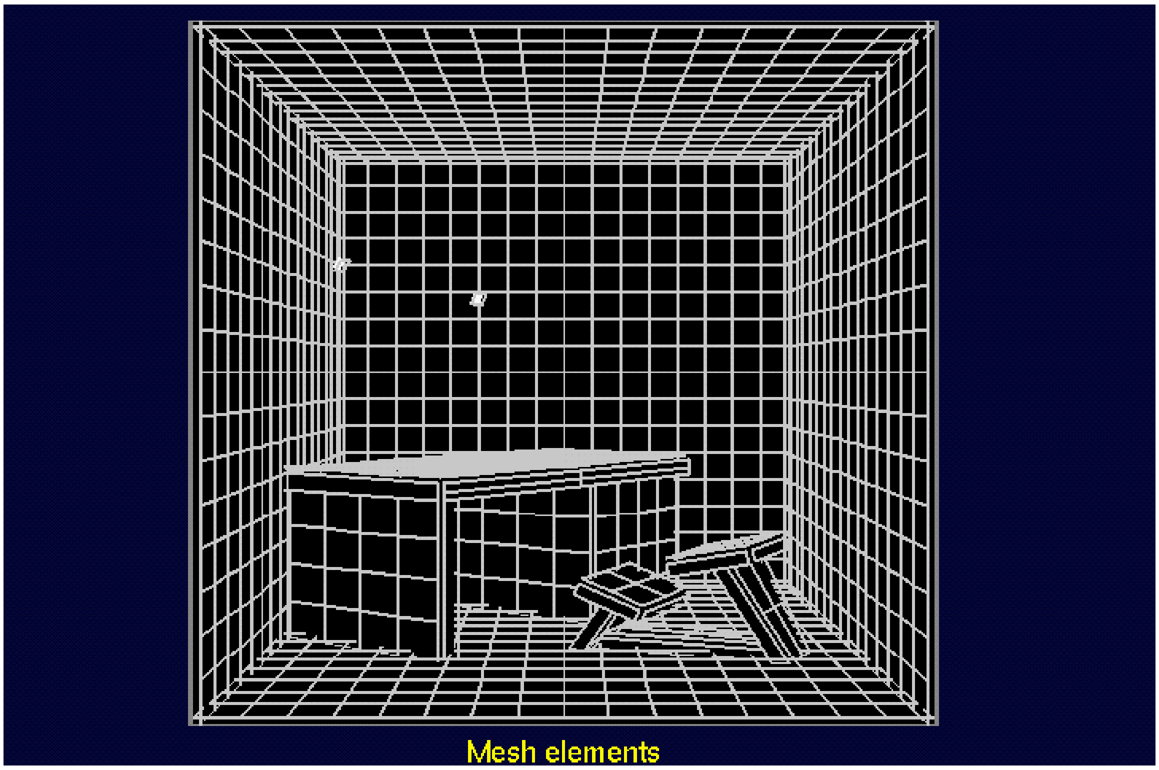


Solving the Radiosity Equation

Iterate system until convergence
- Improve "guess" for each B_i based on the B_j, $j \neq i$
- In practice, can't capture fine-scale illumination effects unless original surfaces are "meshed"

Terminology (somewhat arbitrary):
- Surfaces: boundaries of scene objects
- Patches: surface portion acting as illumination source
- Elements: patch portion receiving illumination

Discretization



Iteration

Numerical iteration of radiosity values

At equilibrium:

$$B_i = E_i + \rho_i \sum_j B_j F_{ij}$$

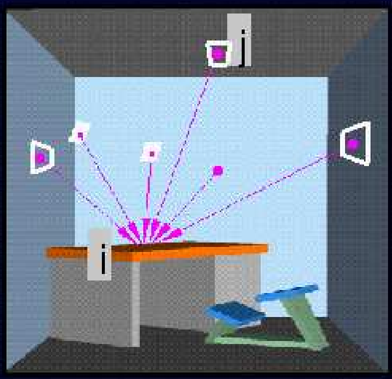
So, given estimates at step t, produce estimates at step $t+1$ as:

$$B_i^{t+1} = E_i + \rho_i \sum_j B_j^t F_{ij}$$

Solving the Radiosity Equation

Radiosity overview

Radiosity computation yields equilibrium light distribution



$$B_i = E_i + \rho_i \sum_{j=1}^{N} B_j F_{ji}$$

where the E_i and ρ_i are given, the B_i are the unknowns, and:

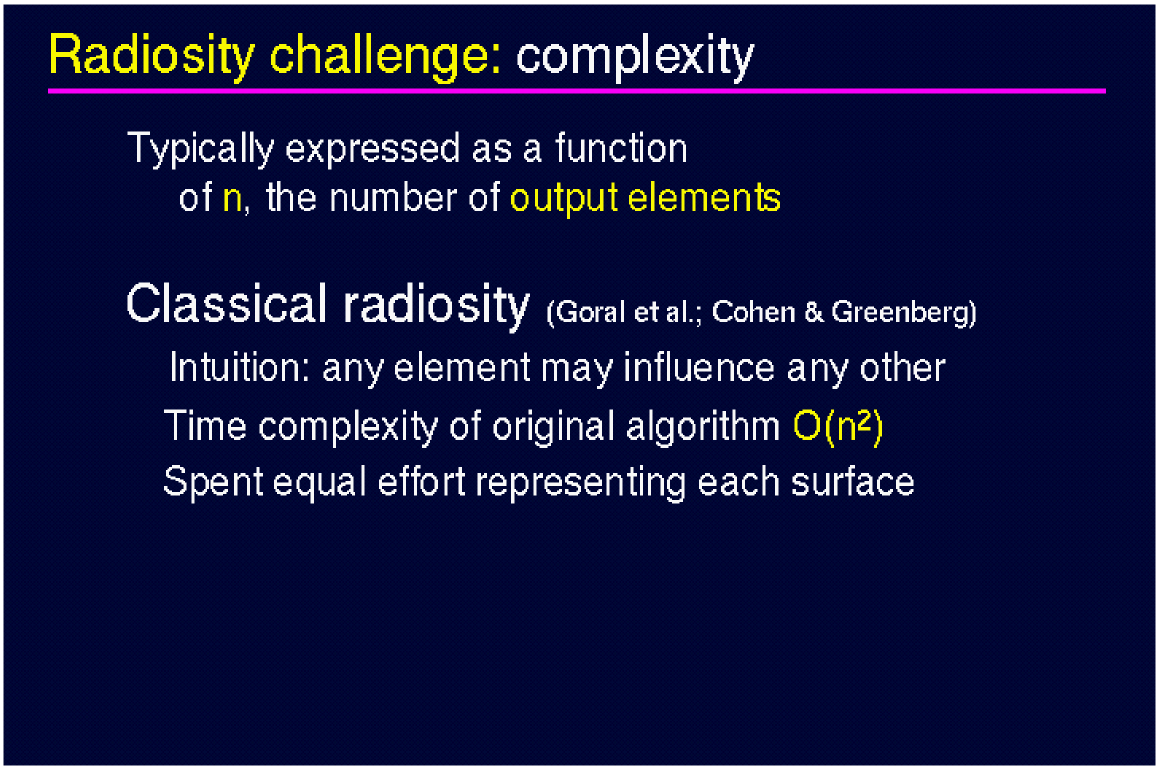
B_i (B_j) = radiant power of element i (j), in watts

E_i = emissive power of element i, in watts

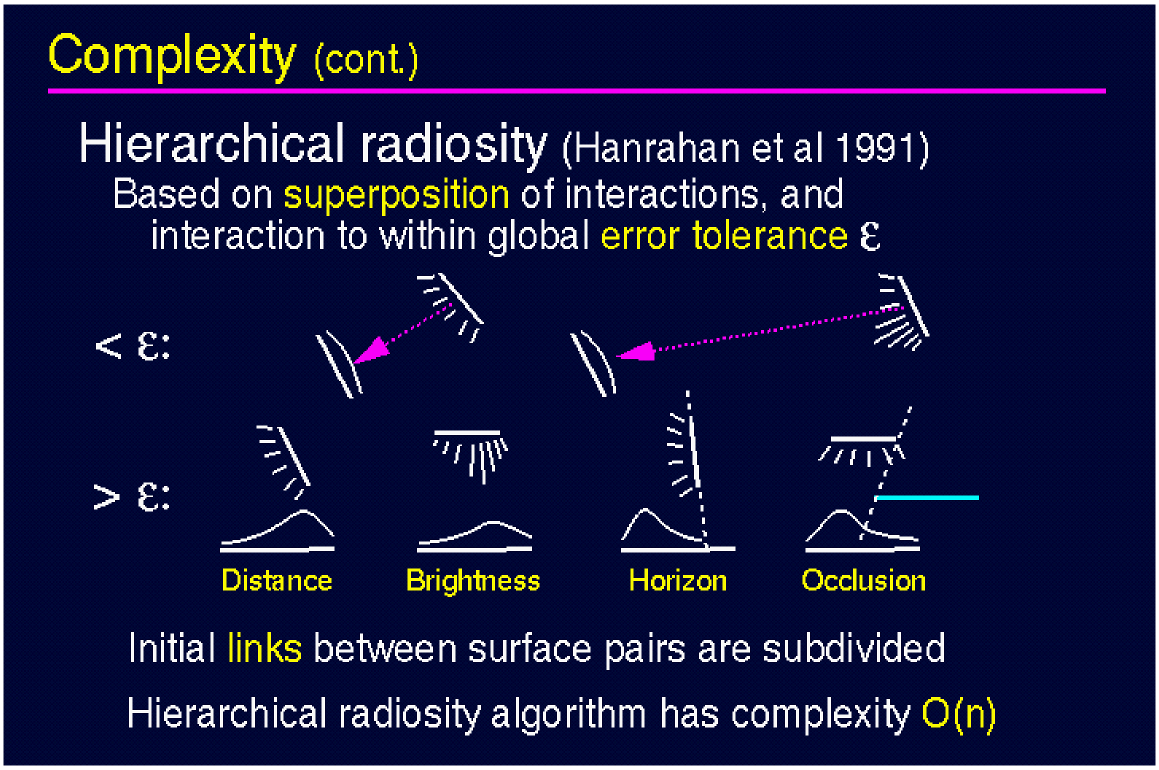
ρ_i = reflectivity of element i, unitless

F_{ji} = fraction of element j's power arriving at i

Asymptotics



Hierarchy



Complexity (cont.)
Hierarchical radiosity (Hanrahan et al 1991)
Based on superposition of interactions, and
interaction to within global error tolerance ε
< ε:
> ε:
Distance
Brightness
Horizon
Occlusion
Initial links between surface pairs are subdivided
Hierarchical radiosity algorithm has complexity O(n)

Resulting Mesh

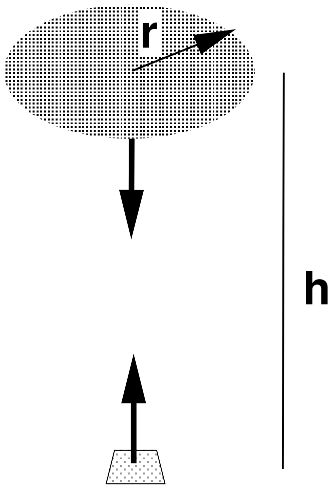
Resulting Solution

Computing Form Factors

What is the form factor? (Intuition.)
  If patches are back-facing?
  If patches are front-facing, but occluded?
  If patches are front-facing, and intervisible?
    What is a good approximation?

Computing VISIBLE **Form Factors**

Approximating front-facing, intervisible F_{ij}



For i differential, j a parallel disk of radius r
Then solid angle subtended by j from i is

$$\frac{r^2}{h^2 + r^2}$$

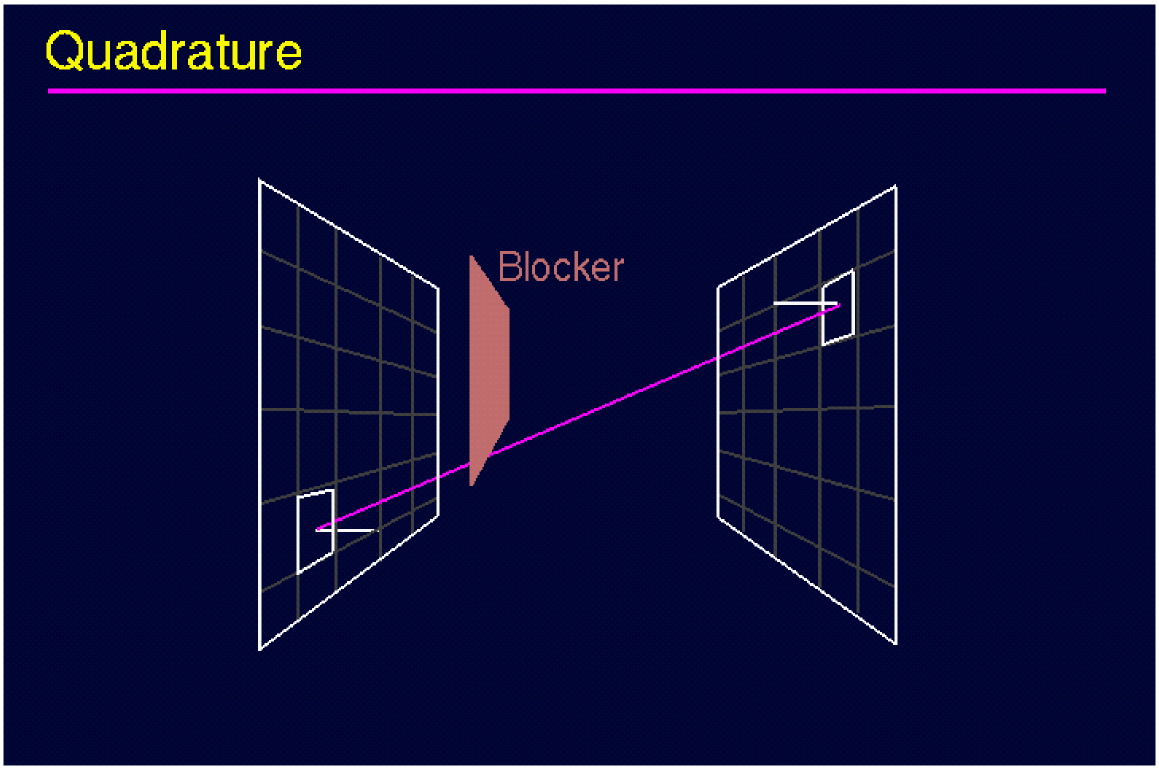
If non-parallel, incorporate a cosine term
Closed-form expressions exist for other cases

Computing PARTIAL **Form Factors**

Front-facing, but only **partially** intervisible?

- Very complex; no known closed-form sol'n
- What to do?

Form Factors with Occlusion



More Complex Solution

Solution Mesh

HKN Questions

Please answer **also**

I am satisfied with Java development and execution environments provided by Athena

(disagree, neutral, agree, etc.)

I did most of my project development on non-Athena machine

(disagree, neutral, agree, etc.)

I found the content of recitation sections valuable

(disagree, neutral, agree, etc.)