

---

# CS580: Radiosity

---

Sung-Eui Yoon  
(윤성익)

Course URL:  
<http://sgvr.kaist.ac.kr/~sungeui/GCG>

# Schedule

---

- **16 students (as of 3/5)**
- **Apr. 25, 27: Students Presentation 1, 2 (2 talks per each class)**
- **May 2, 4: SP 3, 4**
- **May 9, 11: Mid-term project presentation**
- **May 16, 18 Students Presentation 5, 6 (2 talks per each**
- **May 23 25; SP 7, 8**
- **May 30, June 8 Final presentation**
- **June 13, 15 reservation for now (exam period, no class for now)**

# Announcements

---

- **Make a project team of 2 or 3 persons for your final project**
  - **Each student has a clear role**
  - **Declare the team at the KLMS by March-29; you don't need to define the topic by then**
- **Each student**
  - **Present two papers related to the project in each talk**
  - **20 min for each talk; we will have 10 min Q&A**
- **Each team**
  - **Give a mid-term review presentation for the project**
  - **Give the final project presentation**

# Deadlines

---

- **Declare project team members**
  - **By 3/29 at KLMS**
  - **Confirm schedules of paper talks and project talks at 3/30**
  
- **Declare two papers for student presentations**
  - **by 4/10 at KLMS**
  - **Discuss them at the class of 4/11**

# Class Objective (Ch. 11)

---

- **Understand radiosity**
  - Radiosity equation
  - Solving the equation

# History

---

- **Problems with classic ray tracing**
  - Not realistic
  - View-dependent
- **Radiosity (1984)**
  - Global illumination in diffuse scenes
- **Monte Carlo ray tracing (1986)**
  - Global illumination for any environment

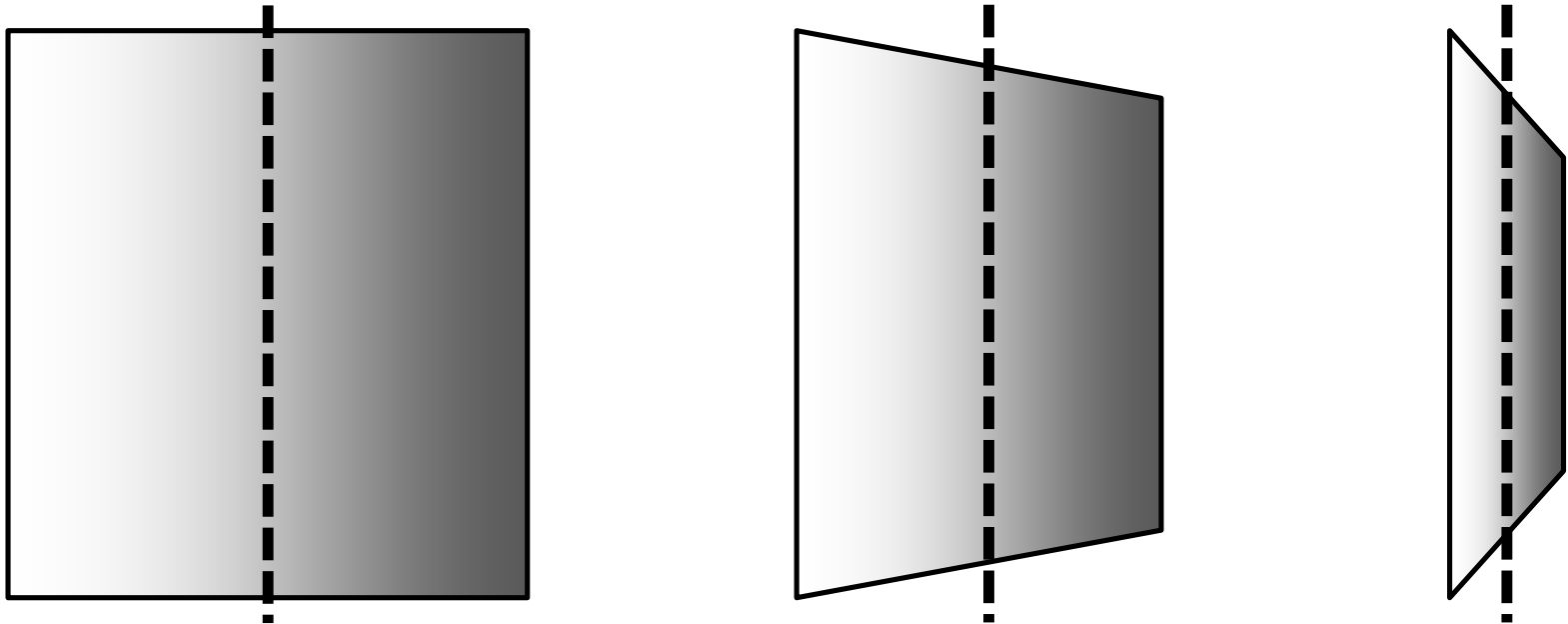
# Radiosity

---

- **Physically based method for diffuse environments**
  - **Support diffuse interactions, color bleeding, indirect lighting and penumbra**
  - **Account for very high percentage of total energy transfer**
  - **Finite element method**

# Key Idea #1: Diffuse Only

---



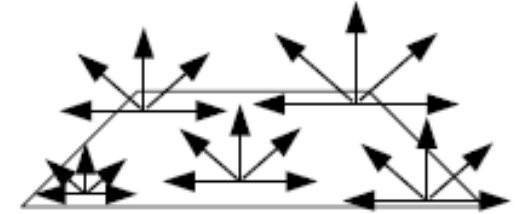
- **Radiance independent of direction**
  - **Surface looks the same from any viewpoint**
  - **No specular reflection**



# Diffuse Surfaces

- **Diffuse emitter**

- $L(x \rightarrow \Theta) = \text{constant over } \Theta$



- **Diffuse reflector**

- **Constant reflectivity**

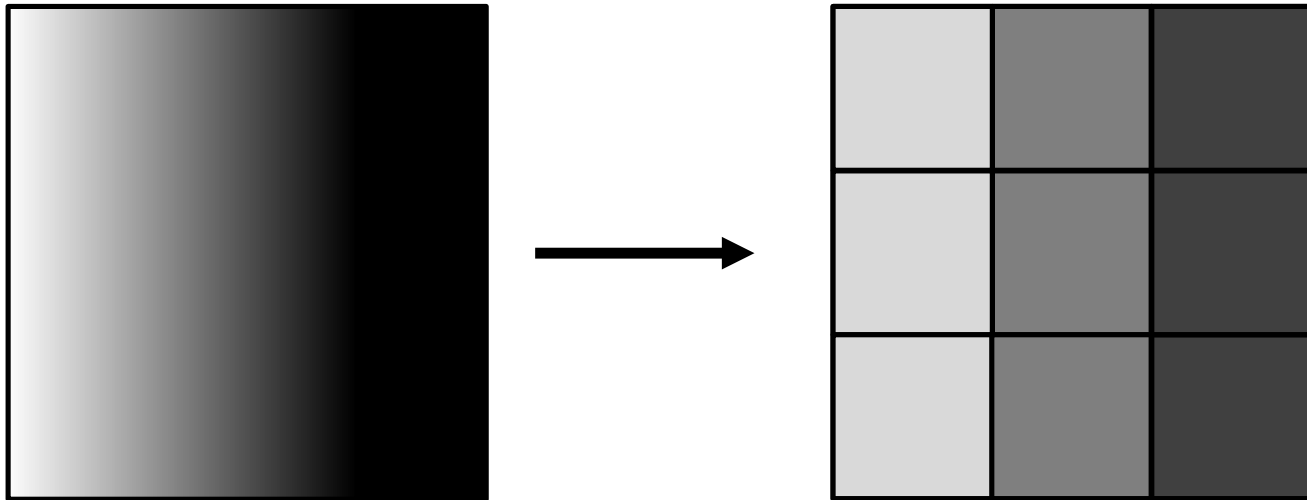


From kavita's slides

# Key Idea #2: Constant Polygons

---

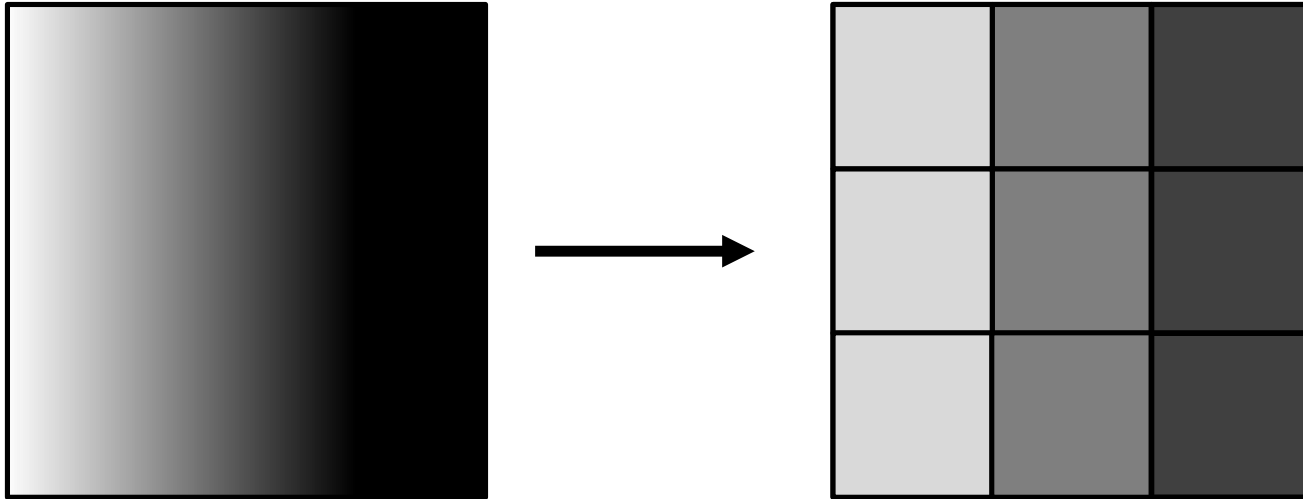
- **Radiosity is an approximation**
  - **Due to discretization of scene into patches**



- **Subdivide scene into small polygons**

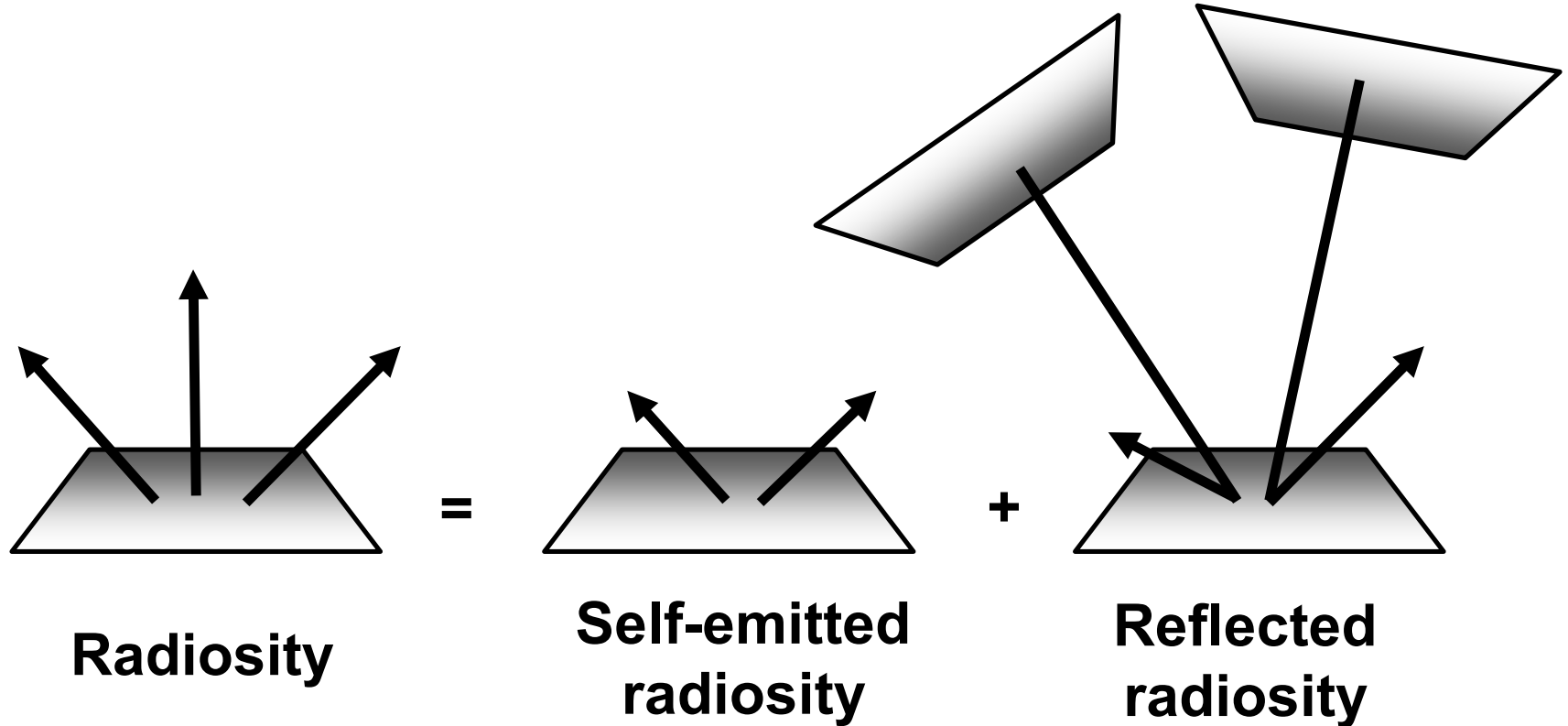
# Constant Radiance Approximation

---



- **Radiance is constant over a surface element**
  - $L(x) = \text{constant over } x$

# Radiosity Equation



$$Radiosity_i = Radiosity_{self,i} + \sum_{j=1}^N a_{j \rightarrow i} Radiosity_j$$

# Radiosity Equations

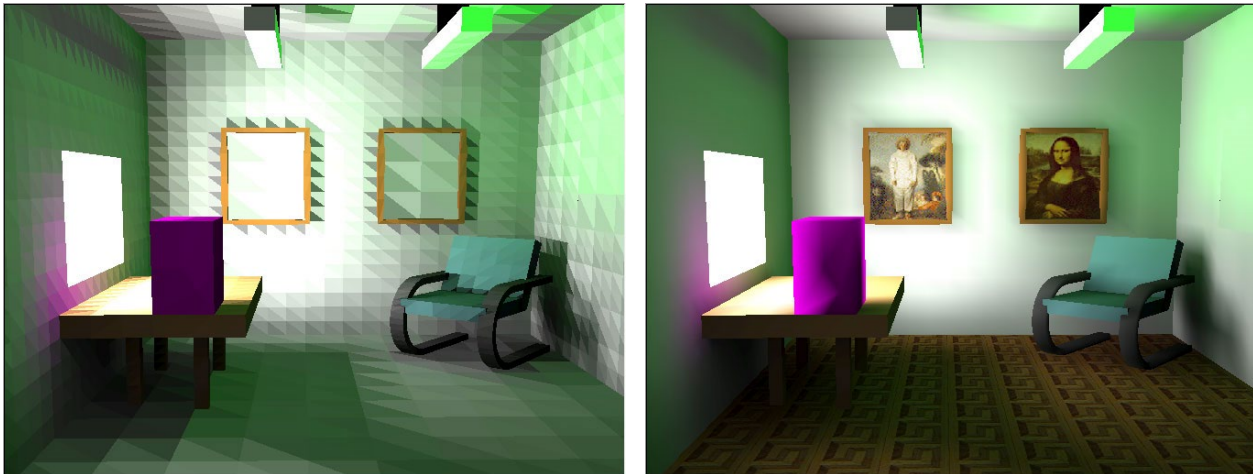
- Radiosity for each polygon  $i$

$$\begin{aligned} \text{Radiosity}_1 &= \text{Radiosity}_{self,1} + \sum_{j=1}^N a_{j \rightarrow 1} \text{Radiosity}_j \\ &\vdots \\ \text{Radiosity}_i &= \text{Radiosity}_{self,i} + \sum_{j=1}^N a_{j \rightarrow i} \text{Radiosity}_j \\ &\vdots \\ \text{Radiosity}_N &= \text{Radiosity}_{self,N} + \sum_{j=1}^N a_{j \rightarrow N} \text{Radiosity}_j \end{aligned}$$

- $N$  equations and  $N$  unknown variables

# Radiosity Algorithm

- **Subdivide the scene in small polygons**
- **Compute a constant illumination value for each polygon**
- **Choose a viewpoint and display the visible polygon**
  - **Keep doing this process**



From Donald Fong's slides

# Radiosity Result

---



# Theatre Scene

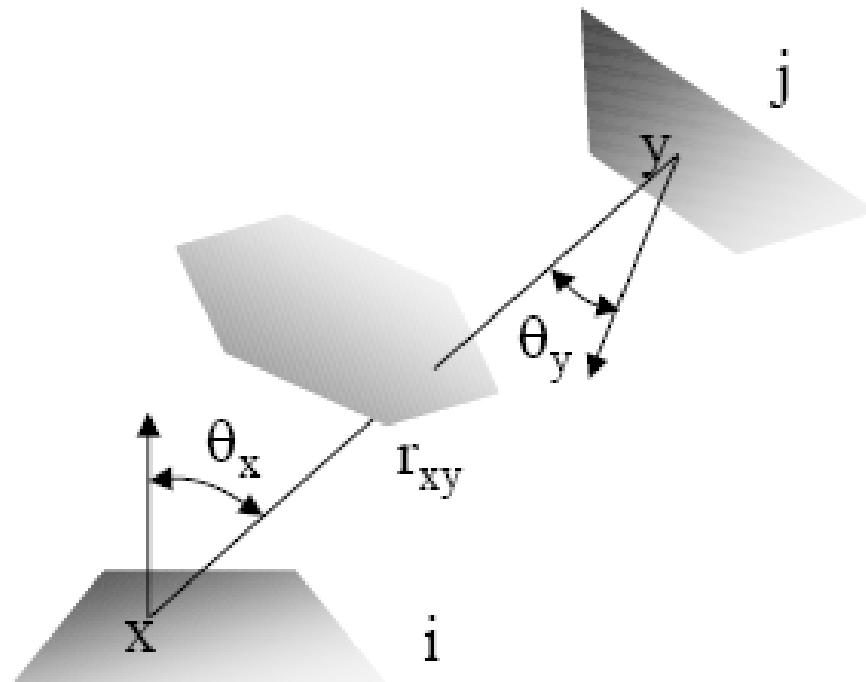
---





# Compute Form Factors

$$F(j \rightarrow i) = \frac{1}{A_j} \int_{A_i} \int_{A_j} \frac{\cos \theta_x \cdot \cos \theta_y}{\pi \cdot r_{xy}^2} \cdot V(x, y) \cdot dA_y \cdot dA_x$$



# Radiosity Equation

---

- **Radiosity for each polygon  $i$**

$$B_i = B_{e,i} + \rho_i \sum_j B_j F(i \rightarrow j)$$

- **Linear system**

- $B_i$  : radiosity of patch  $i$  (unknown)
- $B_{e,i}$  : emission of patch  $i$  (known)
- $\rho_i$  : reflectivity of patch  $i$  (known)
- $F(i \rightarrow j)$ : form-factor (coefficients of matrix)

# Linear System of Radiosity

$$\begin{array}{c} \text{Known} \\ \left[ \begin{array}{cccc} 1 - \rho_1 F(1 \rightarrow 1) & -\rho_1 F(1 \rightarrow 2) & \dots & -\rho_1 F(1 \rightarrow n) \\ \vdots & \vdots & \ddots & \vdots \\ -\rho_n F(n \rightarrow 1) & -\rho_n F(n \rightarrow 2) & \dots & 1 - \rho_n F(n \rightarrow n) \end{array} \right] \end{array} \begin{array}{c} \\ \left[ \begin{array}{c} B_1 \\ \vdots \\ B_n \end{array} \right] \\ \uparrow \\ \text{Unknown} \end{array} = \begin{array}{c} \text{Known} \\ \left[ \begin{array}{c} B_{e,1} \\ \vdots \\ B_{e,n} \end{array} \right] \end{array}$$

# How to Solve Linear System

---

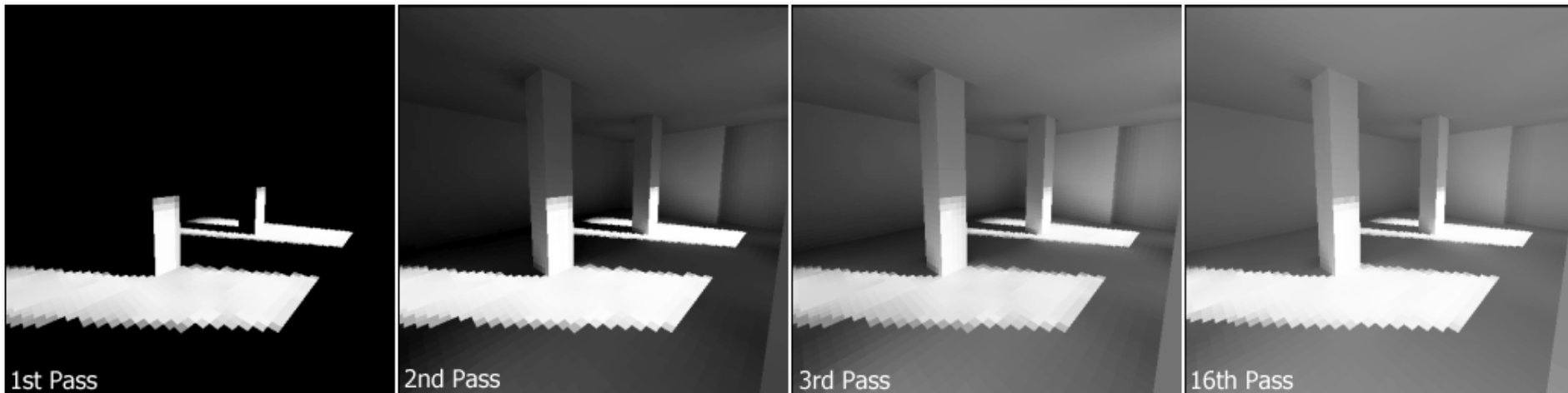
- **Matrix inversion**
  - Takes  $O(n^3)$
- **Gather methods**
  - Jacobi iteration
  - Gauss-Seidel
- **Shooting**
  - Southwell iteration



# Progress of Update Steps

---

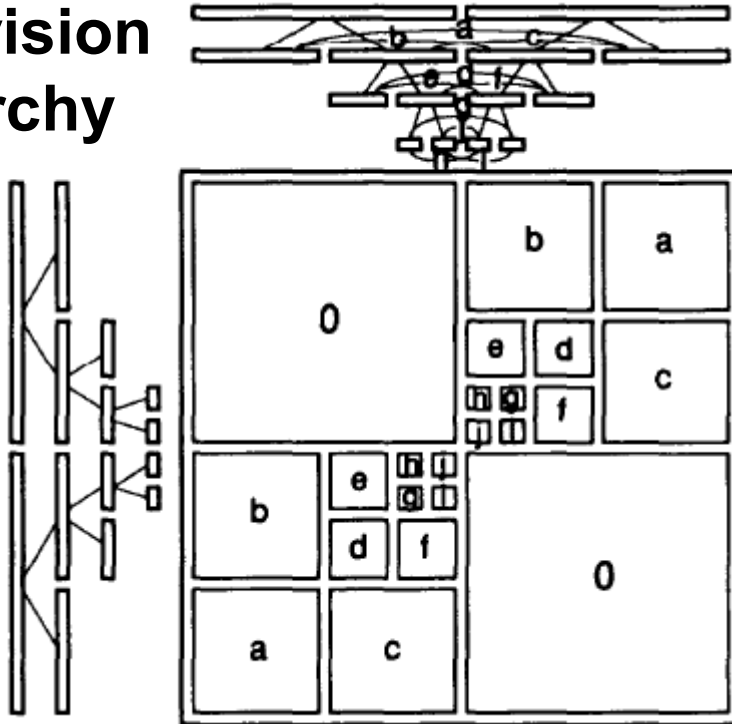
- Update step supports the light bounce



# Multi-Resolution Approach

- **A Rapid Hierarchical Radiosity Algorithm, Hanrahan, et al, SIGGRAPH 1991**

Subdivision hierarchy



- **Refine triangles only if doing so improves the foam factor accuracy above a threshold**

Block diagram of the form factor matrix

# Hybrid and Multipass Methods

---

- **Ray tracing**
  - **Good for specular and refractive indirect illumination**
  - **View-dependent**
- **Radiosity**
  - **Good for diffuse**
  - **Allows interactive rendering**
  - **Does not scale well for massive models**
- **Hybrid methods**
  - **Combine both of them in a way**

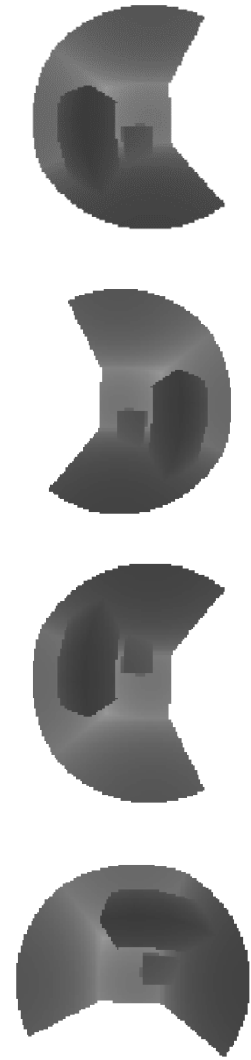
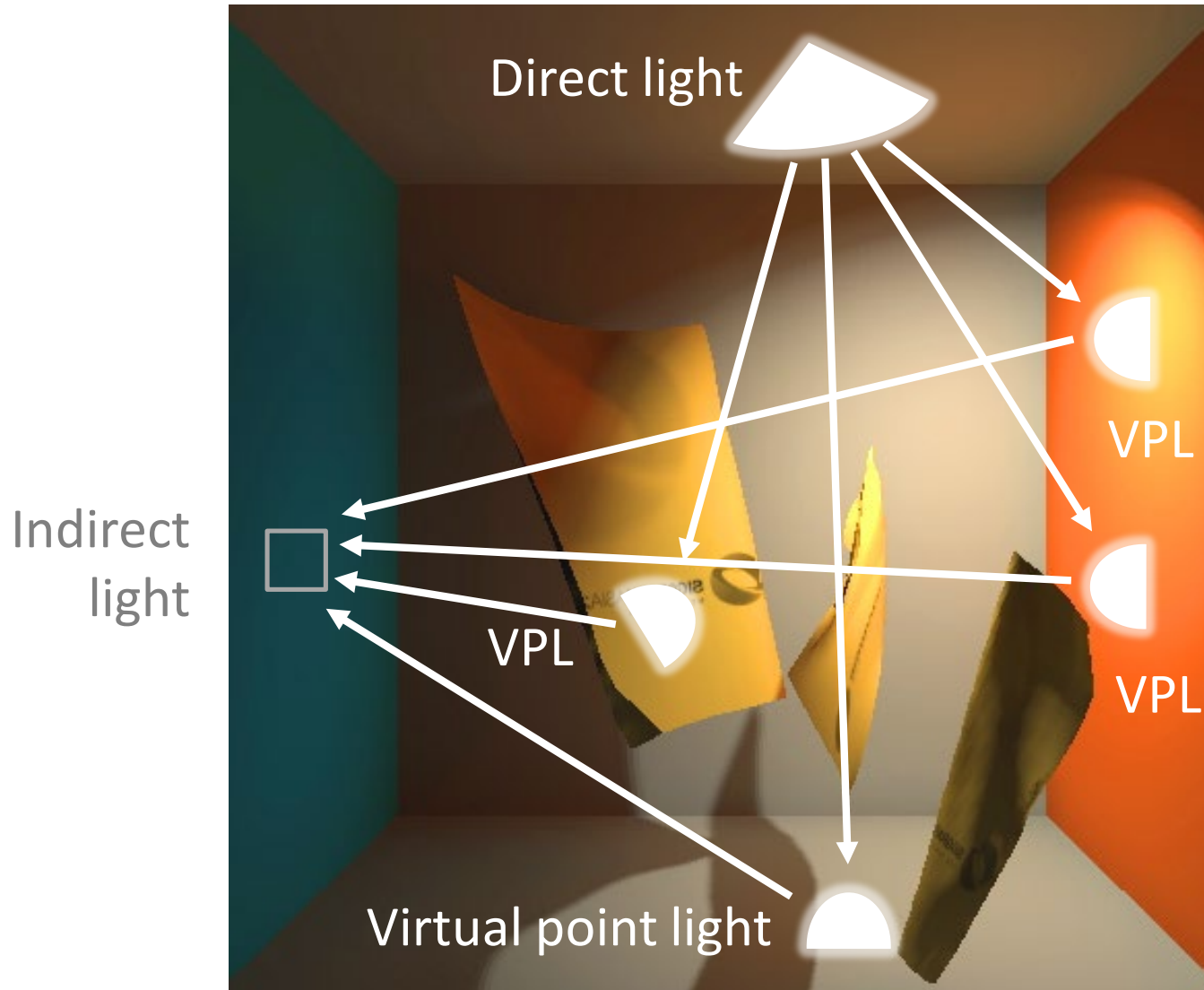


# Instant Radiosity

---

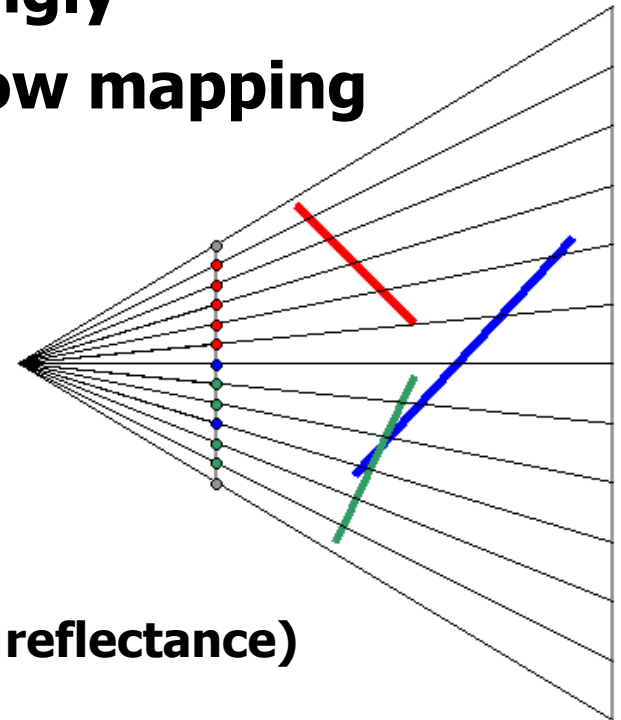
- **Use the concept of radiosity**
- **Map its functions to those of classic rendering pipeline**
  - **Utilize fast GPU**
- **Additional concepts**
  - **Virtual point lights**
  - **Shadow maps**

# Instant Radiosity

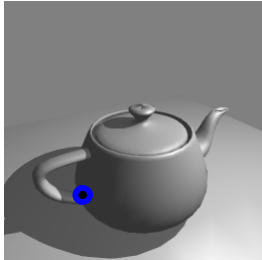


# Ray Casting

- For each pixel, find closest object along the ray and shade pixel accordingly
- Produce similar results to shadow mapping
- Advantages
  - Conceptually simple
  - Can support CSG
  - Can take advantage of spatial coherence in scene
  - Can be extended to handle global illumination effects (ex: shadows and reflectance)
- Disadvantages
  - Renderer must have access to entire retained model
  - Hard to map to special-purpose hardware
  - Visibility computation is a function of resolution

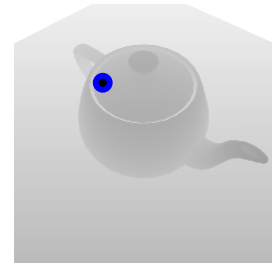


# Shadow Maps

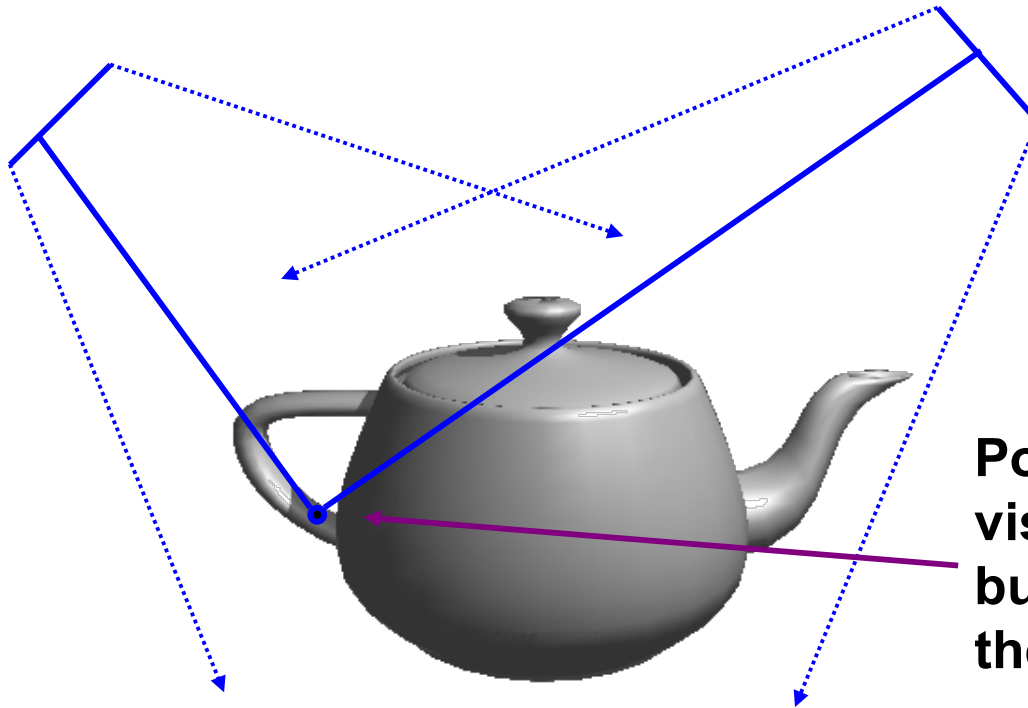


Eye

Use the depth map in the light view to determine if sample point is visible



Light

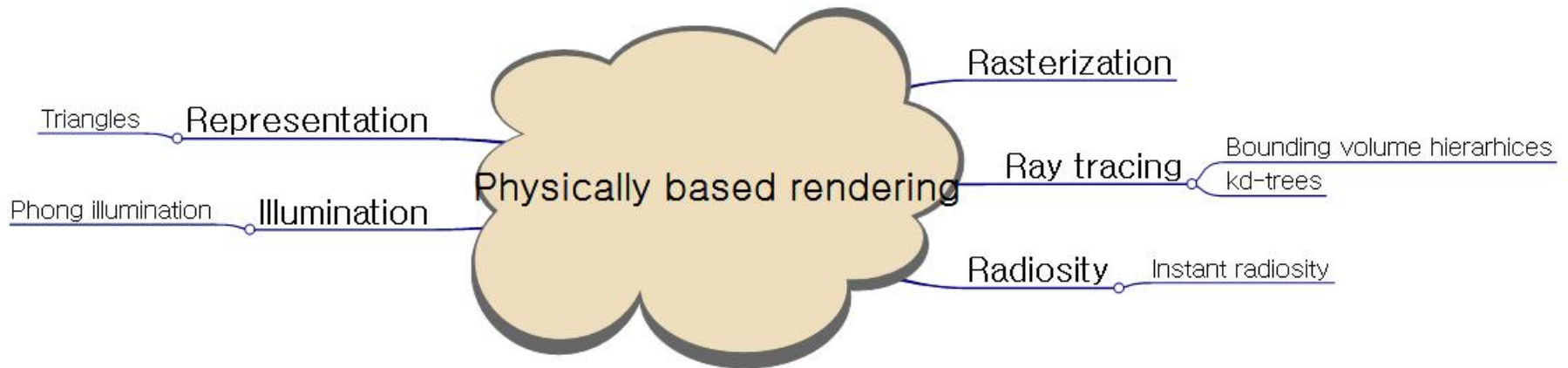


Point in shadow visible to the eye, but not visible to the light

# Class Objectives were:

---

- **Understand radiosity**
  - Radiosity equation
  - Solving the equation



# Homework

---

- **Go over the next lecture slides before the class**
- **Watch 2 paper videos and submit your summaries every Mon. class**
  - **Just one paragraph for each summary**

## Example:

**Title: XXX XXXX XXXX**

**Abstract: this video is about accelerating the performance of ray tracing. To achieve its goal, they design a new technique for reordering rays, since by doing so, they can improve the ray coherence and thus improve the overall performance.**

# Any Questions?

---

- **Submit three times before the mid-term exam**
- **Come up with one question on what we have discussed in the class and submit at the end of the class**
  - **1 for typical questions**
  - **2 for questions that have some thoughts or surprise me**

# Next Time

---

- **Radiometry and rendering equation**