CS580:
Radiosity

Sung-Eui Yoon
(윤성의)

Course URL:
http://sglab.kaist.ac.kr/~sungeui/GCG

Class Objective

- Understand radiosity
  - Radiosity equation
  - Solving the equation

- Related reading
  - Radiosity chapter of the Rendering book

History

- **Problems with classic ray tracing**
  - **Not realistic**
  - **View-dependent**

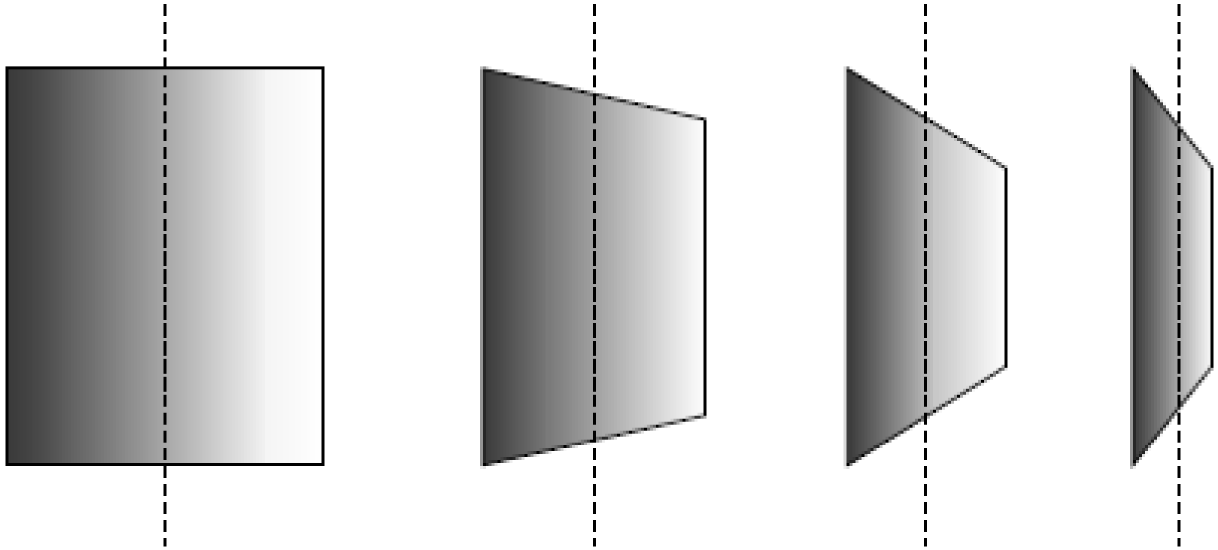
- **Radiosity (1984)**
  - **Global illumination in diffuse scenes**

- **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

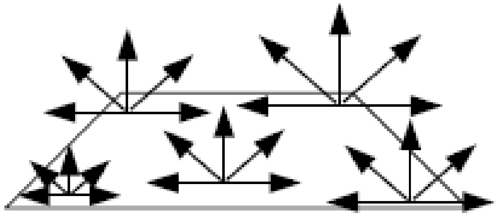


From kavita's slides

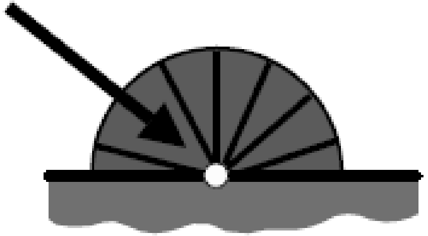
- **Radiance independent of direction**
  - **Surface looks the same from any viewpoint**
  - **No specular reflection**

Diffuse Surfaces

- **Diffuse emitter**
  - $L(x \rightarrow \Theta)$ **= constant over** Θ



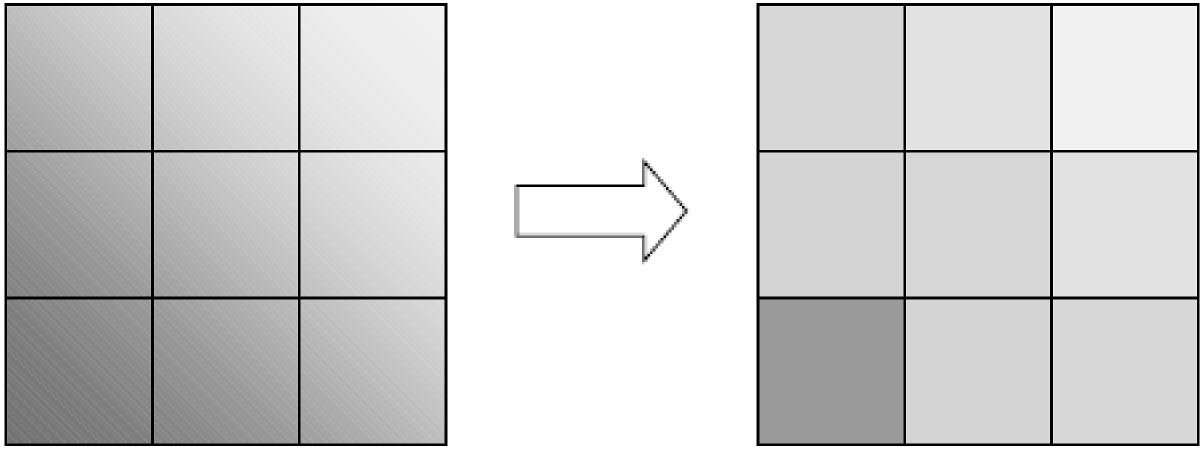
- **Diffuse reflector**
  - **Constant reflectivity**



From kavita's slides

Key Idea #2: Constant Polygons

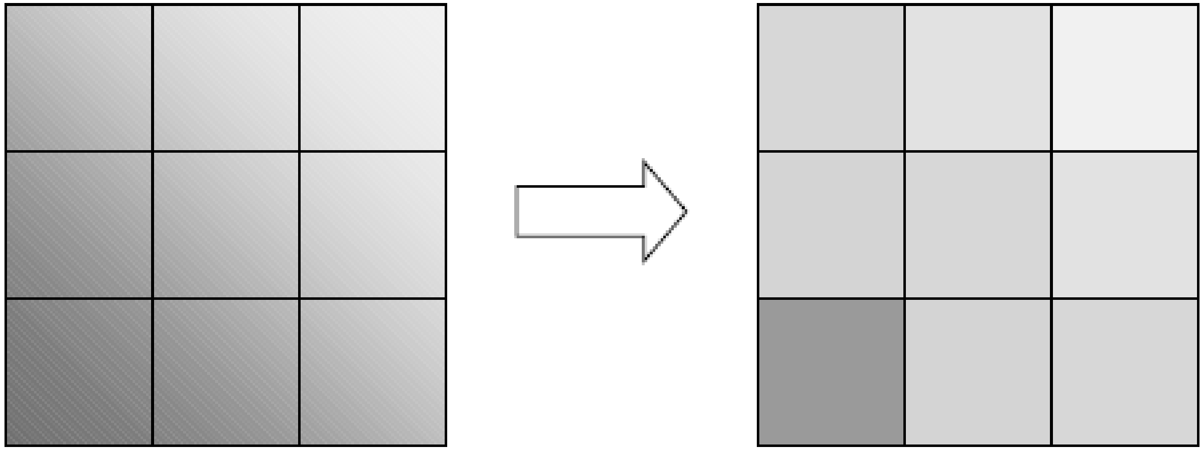
- Radiosity is an approximation
  - Due to discretization of scene into patches



From kavita's slides

  - Subdivide scene into small polygons

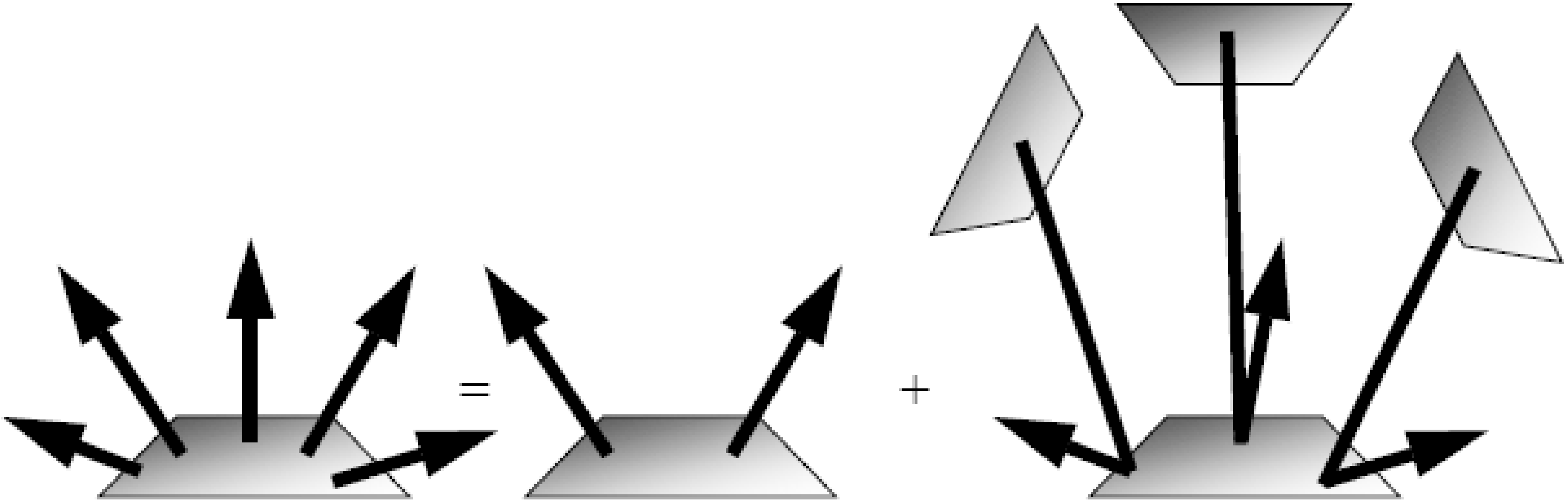
Constant Radiance Approximation



From kavita's slides

- **Radiance is constant over a surface element**
  - **L(x) = constant over x**

Radiosity Equation



Emitted radiosity = self-emitted radiosity + received & reflected radiosity

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

Radiosity Equation

- Radiosity equation for each polygon i

$$Radiosity_1 = Radiosity_{self,1} + \sum_{j=1}^{N} a_{j\to 1} Radiosity_j$$

$$Radiosity_2 = Radiosity_{self,2} + \sum_{j=1}^{N} a_{j\to 2} Radiosity_j$$

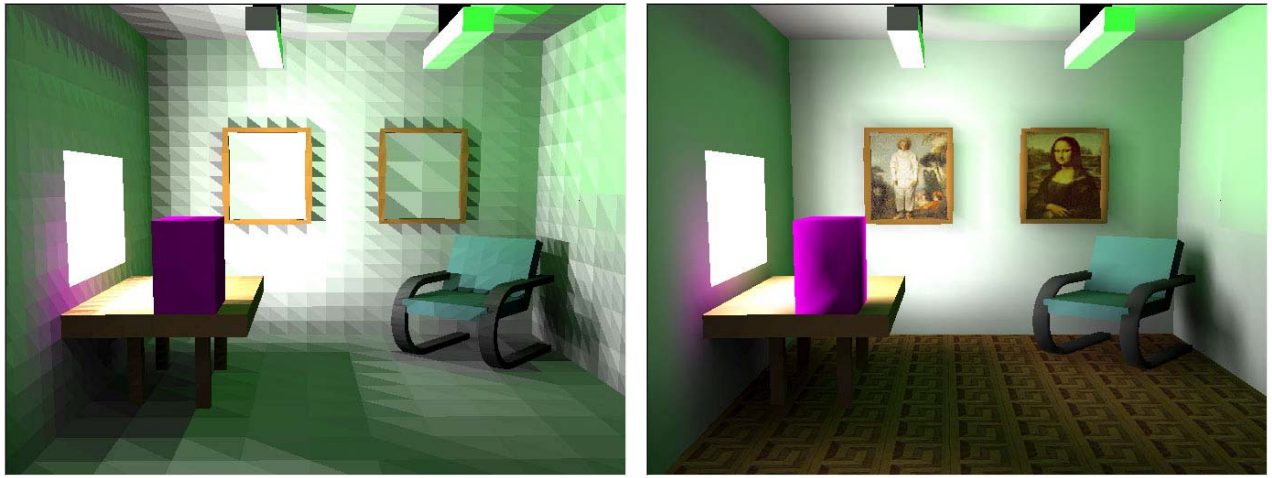
$$\ldots$$

$$Radiosity_N = Radiosity_{self,N} + \sum_{j=1}^{N} a_{j\to N} Radiosity_j$$

- N equations; 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



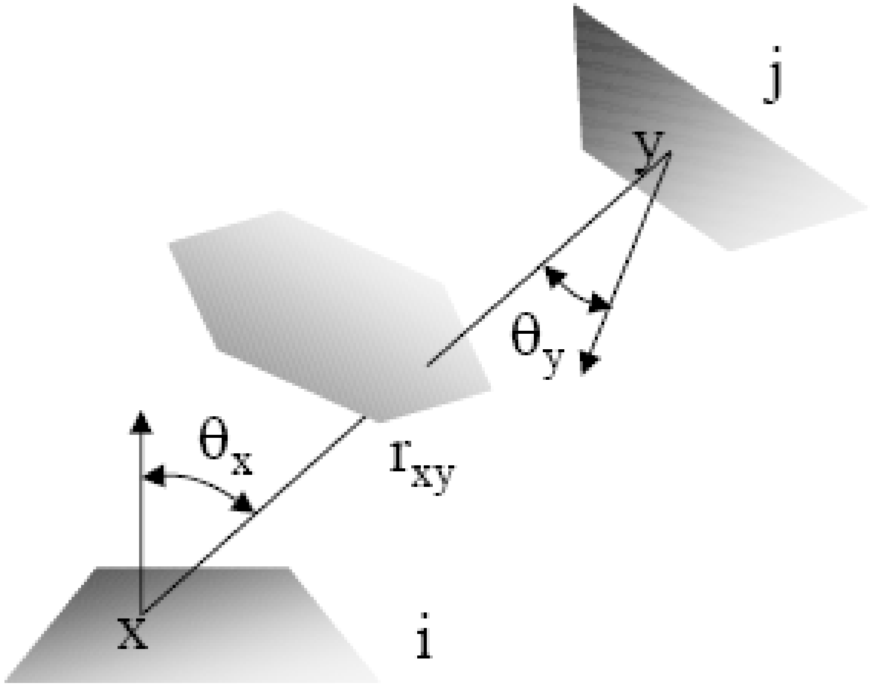
From Donald Fong's slides

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

$$\forall i : B_i = B_{e,i} + \rho_i \sum_{j=1}^{N} B_j F(i \rightarrow j)$$

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

Linear System of Radiosity Equations

$$\begin{bmatrix} 1-\rho_1 F_{1\to 1} & -\rho_1 F_{1\to 2} & \cdots & -\rho_1 F_{1\to n} \\ -\rho_2 F_{2\to 1} & 1-\rho_2 F_{2\to 2} & \cdots & -\rho_2 F_{2\to n} \\ \cdots & \cdots & \cdots & \cdots \\ -\rho_n F_{n\to 1} & -\rho_n F_{n\to 2} & \cdots & 1-\rho_n F_{n\to n} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \cdots \\ B_n \end{bmatrix} = \begin{bmatrix} B_{e,1} \\ B_{e,2} \\ \cdots \\ B_{e,n} \end{bmatrix}$$

known

known

unknown

How to Solve Linear System

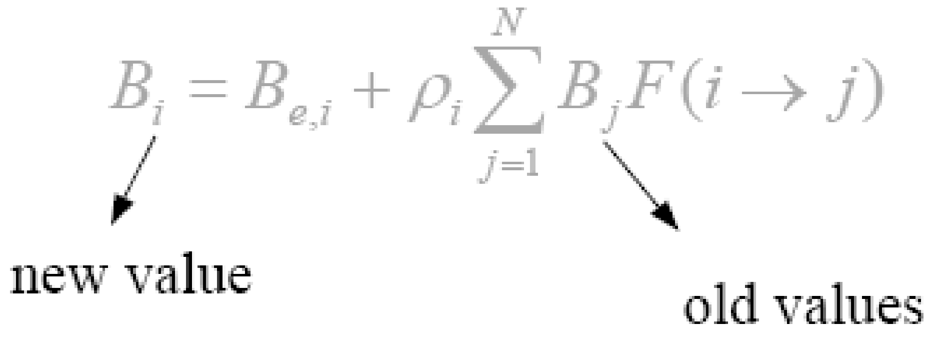
- **Matrix inversion**
  - **Takes O(n^3)**

- **Gather methods**
  - **Jacobi iteration**
  - **Gauss-Seidel**

- **Shooting**
  - **Southwell iteration**

Iterative Approaches

- **Jacobi iteration**
  - **Start with initial guess for energy distribution (light sources)**
  - **Update radiosity of all patches based on the previous guess**

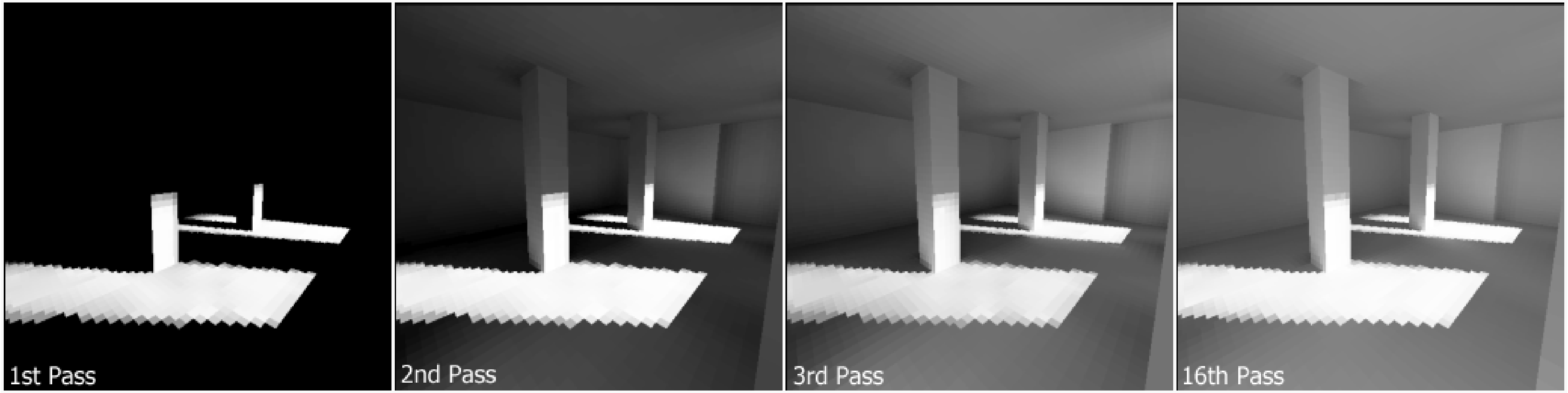


$$B_i = B_{e,i} + \rho_i \sum_{j=1}^{N} B_j F(i \rightarrow j)$$

  - **Repeat until converged**
- **Guass-Seidel iteration**
  - **New values used immediately**

Progress of Update Steps

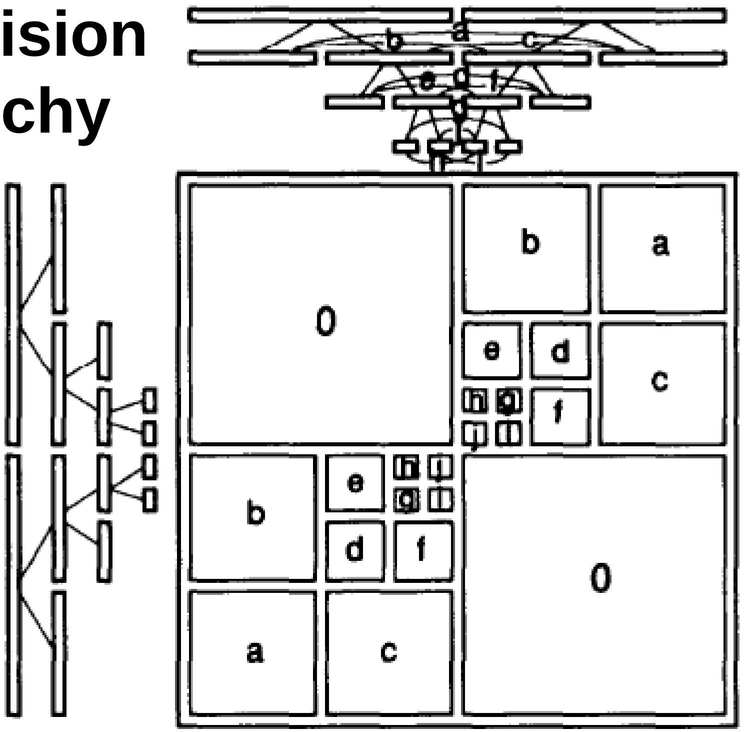
- **Update step supports the light bounce**



Multi-Resolution Approach

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

Subdivision hierarchy



Block diagram of the form factor matrix

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

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

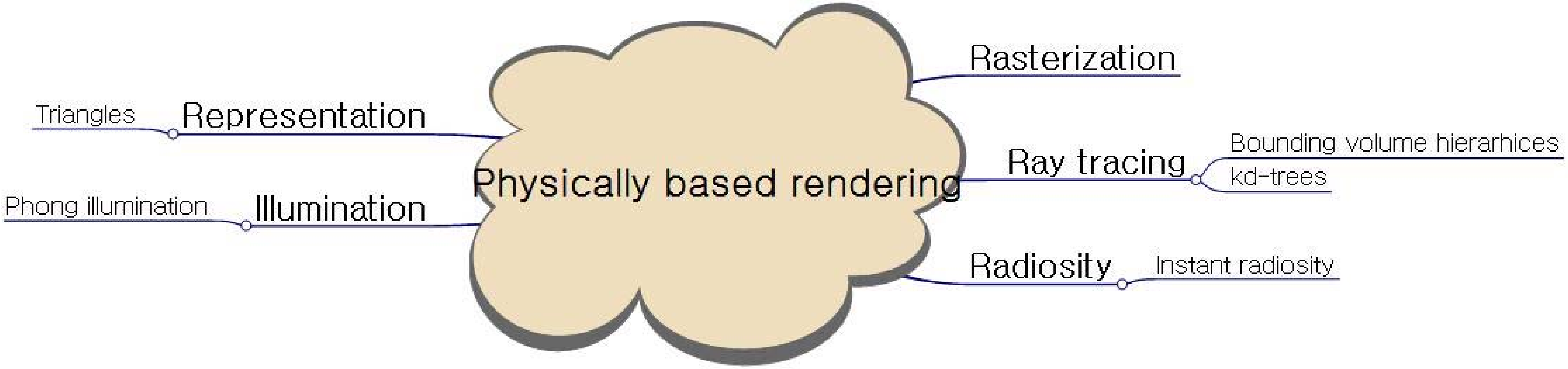
- Hybrid methods
  - Combine both of them in a way

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**
- **Micro-Rendering for Scalable, Parallel Final Gathering (Video)**
  - *Tobias Ritschel, Thomas Engelhardt, Thorsten Grosch, Hans-Peter Seidel, Jan Kautz, Carsten Dachsbacher*
  - ACM Trans. Graph. 28(5) (Proc. SIGGRAPH Asia 2009), 2009.

Class Objectives were:

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



Homework

- **Go over the next lecture slides before the class**
- **Watch 2 SIGGRAPH videos and submit your summaries every Tue. 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?

- Come up with one question on what we have discussed in the class and submit at the end of the class
  - 1 for already answered questions
  - 2 for typical questions
  - 3 for questions with thoughts

- Submit questions at least four times before the mid-term exam
  - Multiple questions for the class is counted as only a single time

Next Time

- **Radiometry**