Light and Shading - Complete Notes

Core Concepts

Light sources emit light (e.g., sun, lamp). Surfaces reflect light depending on their material properties. Reflecting surfaces could be matte (diffuse), shiny (specular), or in-between.

Recognizing Materials

Humans are good at identifying material types based on how they reflect light (e.g., shiny metal vs. rough wood).

Types of Reflections - Diffuse (Lambertian)

Light is scattered equally in all directions. Doesnt depend on the viewers angle.

Lamberts cosine law:

Ld = kd * Il * max(cos(theta), 0) = kd * Il * max(n.l, 0)

Example:

k_d = 0.8, I_l = 10, cos(theta) = 0.866

Ld = 0.8 * 10 * 0.866 = 6.928

Ambient Shading

Represents general background light.

Doesnt depend on angle or direction.

La = ka * Ia

Example:

ka = 0.4, Ia = 8 => La = 3.2

Specular Shading (Blinn-Phong)

Bright spots appear at mirror-reflecting angles. Depends on viewer position.

Ls = Is * ks * cos^n(alpha)

Light and Shading - Complete Notes

Example:

ks = 0.9, Is = 20, cos(alpha) = 0.6, n = 10 => Ls = 20 * 0.9 * (0.6)^10

Flat Shading

One normal vector per polygon. Lighting is computed once per face. Fastest but least realistic. Good for simple shapes.

Gouraud Shading

Normal is averaged per vertex. Lighting is computed at vertices and interpolated across the polygon. Smooth but may miss highlights.

Phong Shading

Interpolates normals per pixel. Computes lighting at each pixel. Most realistic, captures highlights well.

Blinn-Phong Reflection Model

L = La + Ld + Ls = ka * Ia + kd * Il * max(n.l, 0) + ks * Is * cos^n(alpha)

Shading Triangle Meshes

Face shading = Flat, Vertex shading = Gouraud, Pixel shading = Phong

Exam Tip: What Can Appear

- Theory: Differences between shading types, Lambertian model, ambient light importance
- Numerical: Calculate Ld, La, Ls or full lighting with given parameters
- Conceptual: Steps of Phong shading, shading frequency, triangle mesh behavior