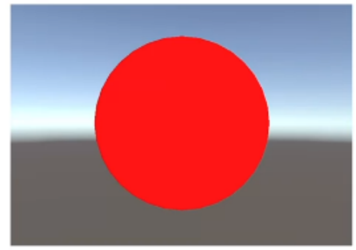
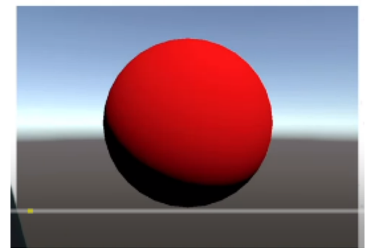
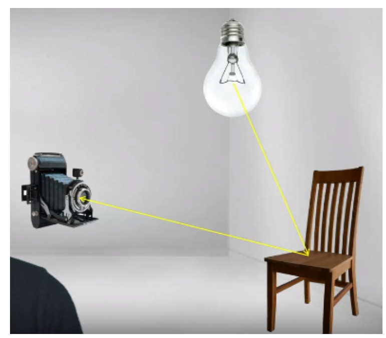
Topic 8 - Lightning and Materials



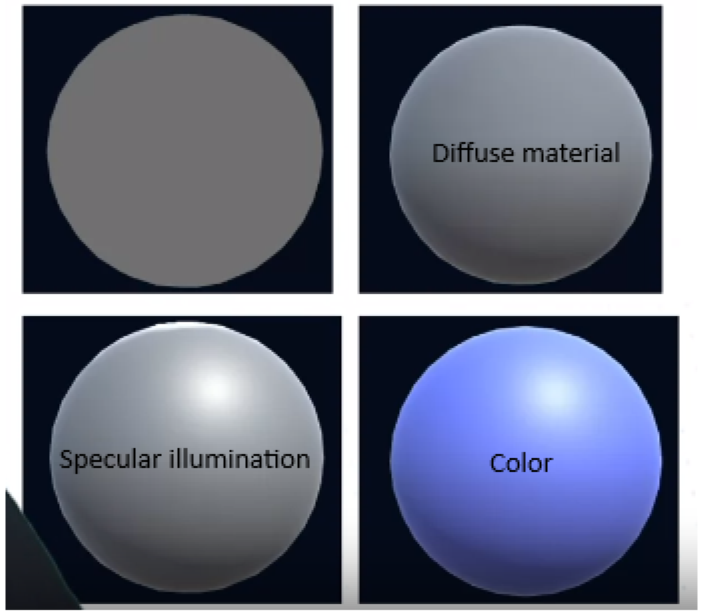




- The way you get a **sense of shape** from an object is that you look at the pattern of lighting and shading across the surface of this object
- The first, most obvious, is to get a sense of lighting, you need **lights**. That's the first thing we're going to learn about this week. But it's also about how the light interacts with the properties of the surface, something we call **materials** in 3D graphics.

So how does lighting work in 3D graphics?

- What we want to know is where does the light come from the light source? Where does it hit the object? And how does it bounce back to the camera?
- We typically have a bunch of different types of **lights**
  - **Directional Lightning**
    - Sun lights
  - **Point Light**
    - Lamp
  - **Spotlight**
    - Cone effect
  - **Ambient light**
    - Apply a certain amount of light to everything wherever it is
- **Materials**:
  - So some objects are shiny that reflect lots of light.
  - It's **what most of what the shader does**
  - material normally means the **interaction of light with the surface**
  - **Diffuse material:**
    - Light is bouncing off the object and then bounces in every direction
    - Diffuse lighting **does not depend on the angle** at which you're looking at an object
  - **Specular illumination:**
    - It's much more direct illumination. The lighting coming off a **mirror** bounces off in exactly the same direction or sort of exactly the same angle at which it bounces back but in the opposite direction. And that means that you can actually see things in a mirror because it's so exact.
    - Spectacular illumination **depend** on the **angle** at which you're looking at an object
  - **color:**
    - A color is an object that only reflects certain wavelengths of light.
    - A blue object will actually absorb all the green and the red wavelengths and only reflect the blue objects



Mathematics of the lightning equation

- a relatively simple way to display something relatively realistic. In order to do that, we need to consider at least three kinds of lighting: **ambient, diffuse, and specular**.

Ambient light

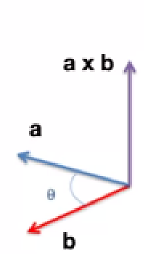
- **Ambient light** is a way to give a bit of color to odd objects
- Ambient light is a basic way to approximate global illumination
- Ambient light is **constant** across the whole object
- We normally set an **ambient light value** for the whole scene(**I_a**)
- And depending on the material of the object, each object will be assigned an **ambient light parameter** (**K_a**)
- So the way to calculate color for a pixel will be Ir equals K_a times I_a, where Ka is the **parameter of the object this pixel is assigned to during the rasterization process**.

  $I_r = k_a I_a$

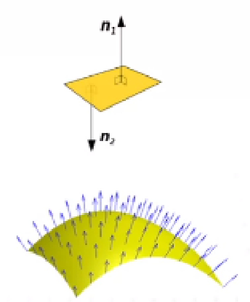
Shading

- When a 3D model is rendered with just color but no shading, it is difficult or impossible to make out its depth and form
- In the real world, the **interaction of light and surfaces give shading**, which humans use as an important depth cue.
- Real shade depends on:
  - Light source

- Material properties
- Viewing angle
- Surface Orientation.
  - We can determine surface orientation using surface normals.
  - A surface normal to a flat surface is a vector that is perpendicular to that surface.
  - In 3D graphics, a normal is an imaginary line that is perpendicular to the surface of a polygon.
  - The normal is used to determine a surface orientation towards a light source.
  - For polygon, such as a triangle, a surface normal can be calculated as the vector cross product of two edges of the polygon.
  - Surface normals are useful in calculating shading, and actually, we calculate two types of shading separately; **diffuse reflection and specular reflection.**

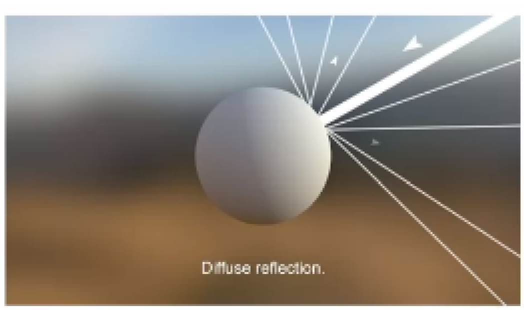


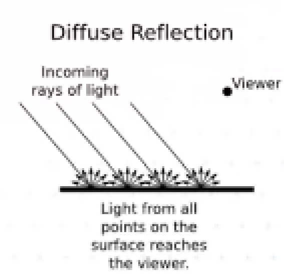
$$\mathbf{a} \times \mathbf{b} = \|\mathbf{a}\| \, \|\mathbf{b}\| \sin\theta \; \mathbf{n}$$



Diffuse Reflection

- Diffuse reflection is a result of lighting interacting with a light scattering surface.
- Think about a ping pong ball or a tennis ball. As we can see with this image, the light comes from the right-hand side. The right side of the ball, which is facing the light source, is well lit, and it gradually gets darker as the surface faces away from the light source.





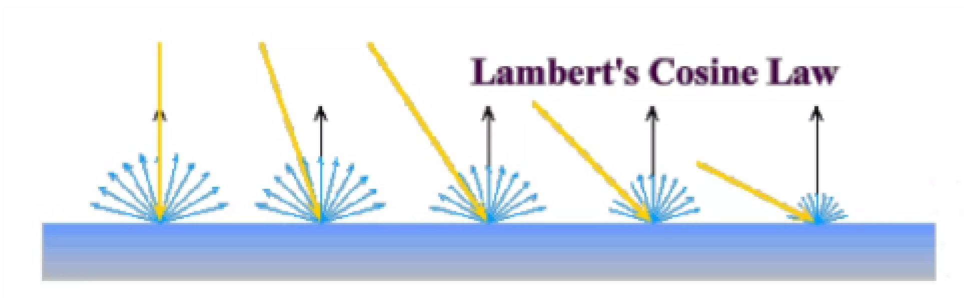
http://www.pernroth.nu/lightandmaterials/

- **How to calculate**
  - You need the **intensity of the light**

- And you need **Angle** θ (which is the angle between point to light ray and surface normal)

Lambert’s cosine law

- Point B receives more light than B. This relationship is described by Lambert's cosine law where the intensity of the light at a certain point is proportional to the cosine value of the angle between the surface normal and a light ray.
- The diffuse reflection contribution should be calculated as **k_d**, which is the **coefficient of the surface material**, times **I_i**, which is the **intensity of the light source**, and cosine Theta, the angle between the surface normal and the light ray.
- Here, we can also replace cosine Theta with the **dot-product** of the two **unit vectors, N and L.** N here is a surface normal and L is the light ray.
- k_d is the **coefficient of the material** of the object. The **bigger k_d** is, the **bigger percentage of lights the surface reflects**.

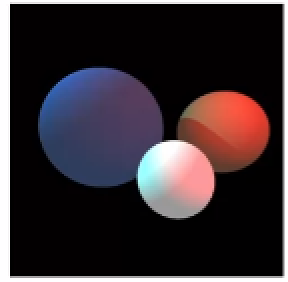


$$k_d I_i cos(\theta) \qquad k_d I_i (n \bullet l)$$

Lightning equation

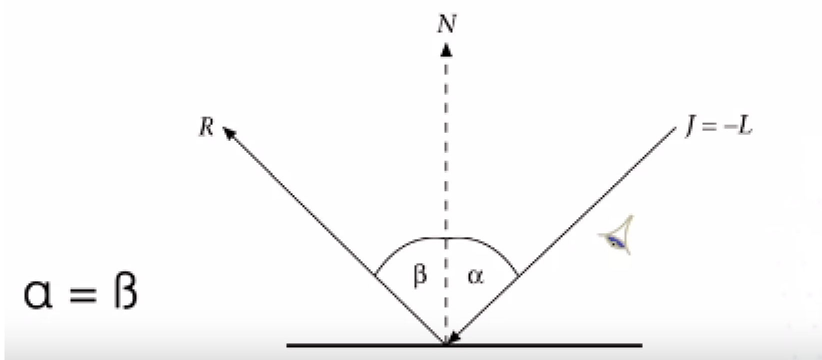
- Ambient (k_a I_a) and Diffuse components (k_d I_i (n*l))

$$I_r = k_a I_a + k_d I_i (n \cdot l)$$



- But the image still looks a bit boring as it lacks the shininess of real-world objects. In order to have the shininess of highlights, we need to also include specular shading.

Specular Shading



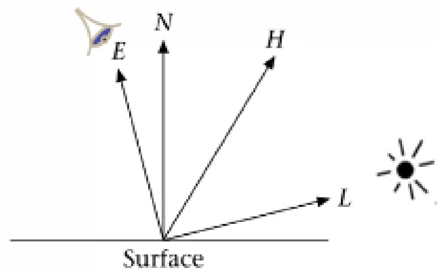
- In real life, many surfaces can be considered shiny. For instance, metal, or very smooth plastic and glasses
- Theoretically, perfectly specular surfaces reflect rays only visible at a certain position.
- The specular shading used in computer graphics is an approximation of specular reflection.
- In fact, we assume all surfaces are glossy instead of perfectly smooth. We use a parameter to define the glossiness
- **How to calculate**, we need to know **three vectors**.
    - N: surface normal
    - L: direction to the light
    - E: direction to the eye

    $$h = \frac{e+l}{||e+l||}$$

    H: a vector that bisects L and E

$$k_s I_i (h \bullet n)^m$$

m: shininess (positive number)



- The same applies to k_s, the bigger the k_s is, the more lights an object reflects
- How does the shininess, **m**, change the effect? Because the maximum value of h dot n is 1, this guarantees that the middle point is always highlighted, but as the point of interest shifts away from the center, where the pure specular effects apply, h dot n will be getting closer to zero, where the highlight vanishes. The higher m is, the more quickly this highlight effect vanishes, the smoother the highlight point is. The bigger m will give a more specular looking surface.