More Lights in Computer Graphics

Keagan Edwards, Dion Udokop

Outline

- Colored light sources
- Ambient light
- Spot light sources
- Area light sources
- Dual-layer light sources

Colored Lights

- So far in this class, all of our lights have been white (or rather, grayscale)
- What if we want colored lights?
- Here is our diffuse shading equation:

$$L_d = k_d I \max(0, \vec{n} \cdot \vec{\ell})$$

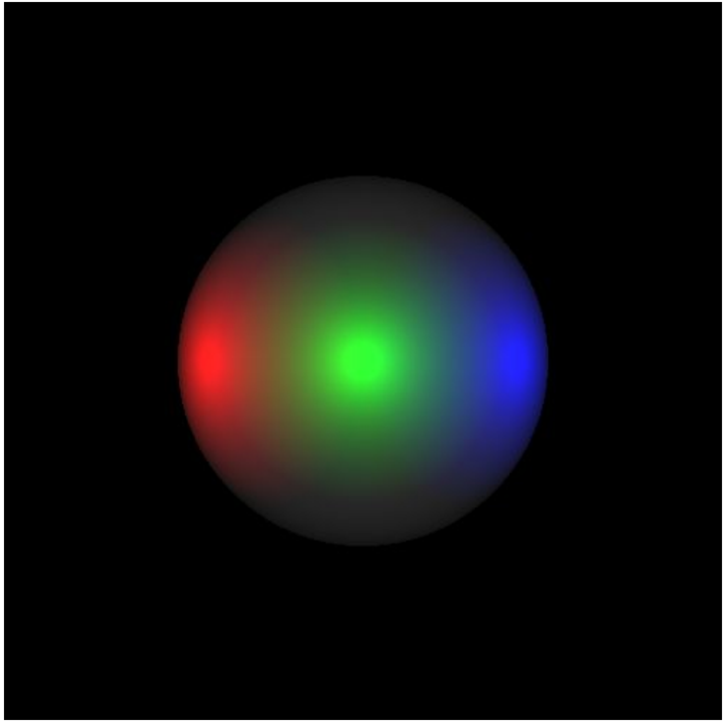
- **Think-Pair-Share:** How can we change it to support colored lighting?

Colored Lights

- Make our light intensity I an RGB triple instead of a scalar
- $I = [I_r, I_g, I_b]$
- We multiple it by the diffuse factor elementwise

$$L_d = k_d I \max(0, \vec{n} \cdot \vec{\ell})$$

Colored Lights



- Single sphere is illuminated in several different colors using colored lights
- This example uses spotlights which we will get to

Ambient Light

- In reality, light bounces over & over, so there is some light everywhere
- For example, you can see underneath a car despite its shadows
- Whereas in a2, shadows were completely black
- Global illumination quickly becomes unreasonable to simulate

Ambient Light

- Hacky solution: everything reflects a small amount of ambient light, regardless of light sources

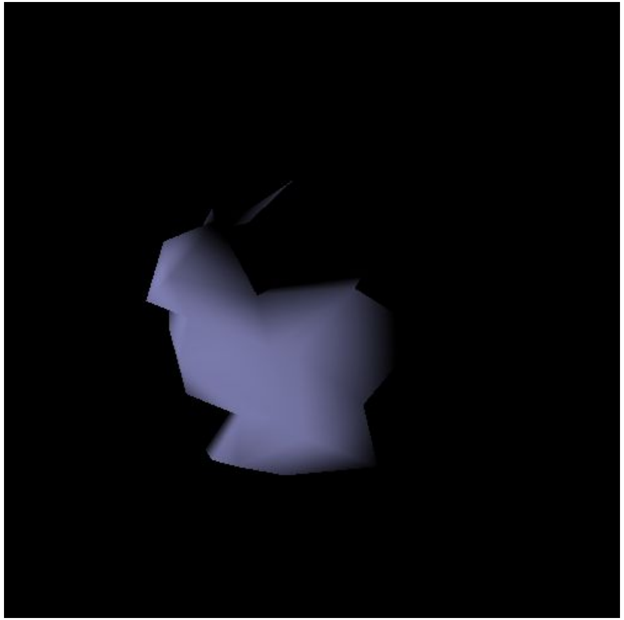
$$L_r = k_a I$$

- We add this ambient light to our existing shading model
- $\mathrm{k_a}$ is our ambient reflection coefficient (can just be $\mathrm{k_d}$)
- I is our ambient light intensity (can also be colored)

Ambient Light

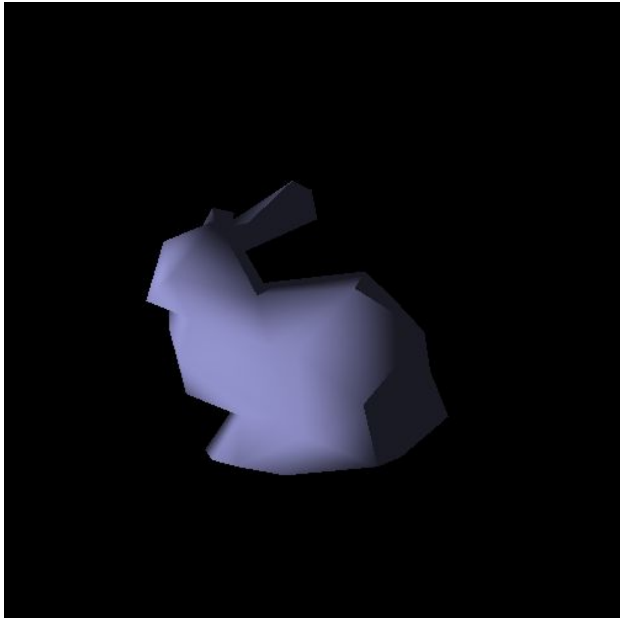
- Ambient light is beneficial because it allows us to see parts of an object that would otherwise be completely black without the performance impact of global illumination
- Doesn’t have to be universally applied to all objects

Ambient Light



Point Light

Ambient Light



Point Light + Ambient Light

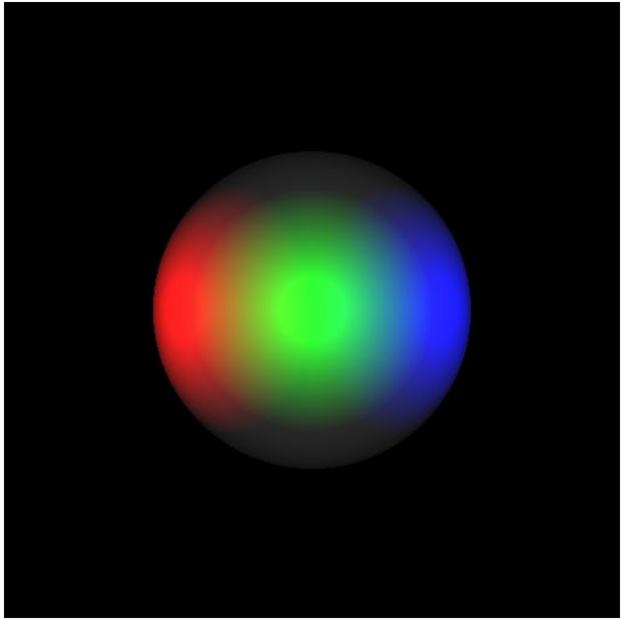
Spot Light

- Simply a point source limited by a cone of angle θ.
- Spotlight is specified by position, intensity, direction, and a falloff factor.
    - The falloff factor determines how quickly the light dims as you move from the center of the light cone toward its edges.
- Implementing spotlight effect:
    - Assuming α is the angle between the spotlight direction and the light direction
    - Where light direction is the vector from the fragment to the light source.
    - If the light is pointing in a direction with intensity I, then the intensity falls off with:

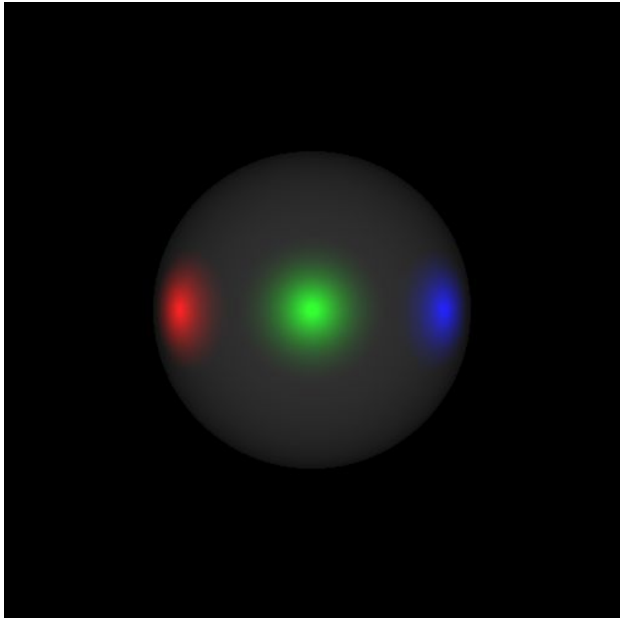
$$I = I_0 \cos^e(\alpha) \quad \text{if } \alpha \leq \theta \text{ otherwise zero}$$

    - The exponent, e, controls the falloff rate.

Spot Light Softness

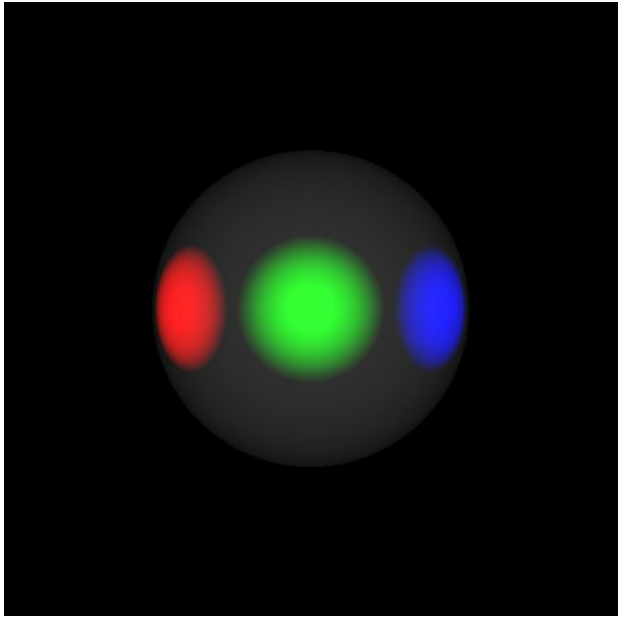


Point Lights



Soft Spot Lights

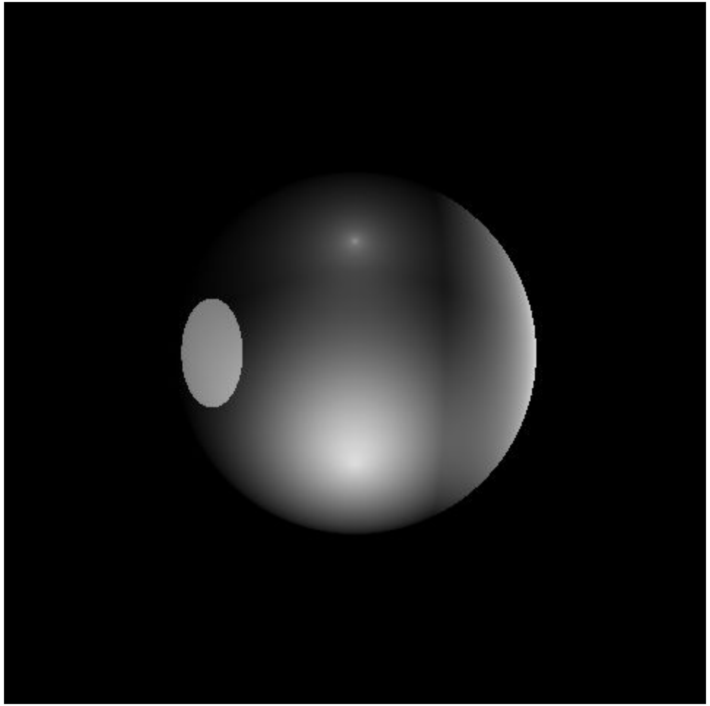
(lower falloff exponent)



Hard Spot Lights

(higher falloff exponent)

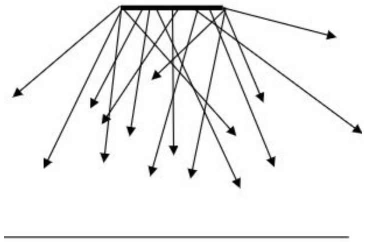
Spot Light

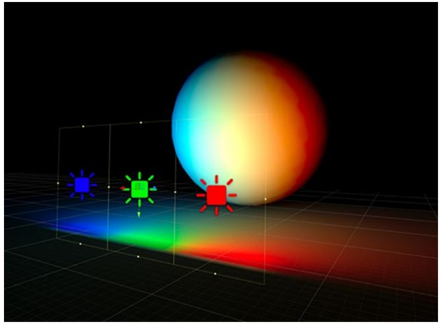


- Spotlights are really flexible
- Top light is really soft
- Left light has no softness
- Right light shows how spot lights can used similar to directional lights

Area Lights

- Light emanates from a surface rather than a point.
  - Usually a rectangle or disc-shaped source.
- The location of the light is not unique.
  - Light is emitted from every point across the shape's surface area.





  - This results in soft shadows and elongated highlights rather than sharp points.

Area Lights

- Problem: Calculating how much of the light from the light source reaches the surface of the object.
- **Think-Pair-Share:** Why is this calculation a problem?
- Infinite number of points on the light source.
    - Impractical.
    - Computationally expensive.
- Solution: Monte Carlo Integration
    - Pick a number of random sample points on the light's surface.
    - Calculate how much light each sample point contributes.
    - Average the contributions and scale by the total area of the light.

Area Lights: Monte Carlo Integration

$$L_d(\mathbf{x},\omega_o) \approx L_i \frac{A}{N} \sum_{k=1}^{N} f(\mathbf{x},\omega_i(k),\omega_o)\, G(\mathbf{x},\mathbf{x'}_k)\, V(\mathbf{x},\mathbf{x'}_k)$$

- $L_d(x,\omega_o)$ → amount of light leaving a point x in the direction towards the camera/viewer
- L_i → light intensity
- A → total area of the light source
- N → number of samples
- Summation Σ from k=1 to N → evaluate the samples and add them up

Area Lights: Monte Carlo Integration

$$L_d(\mathbf{x},\omega_o) \approx L_i \frac{A}{N} \sum_{k=1}^{N} f(\mathbf{x},\omega_i(k),\omega_o)\, G(\mathbf{x},\mathbf{x'}_k)\, V(\mathbf{x},\mathbf{x'}_k)$$

- $f(x,\omega_i,\omega_o)$ → BRDF (Bidirectional Reflectance Distribution Function)
    - BRDF is a function that defines how light from a source is reflected off a surface.
    - x - surface point
    - ω_i - direction the light is coming from (sampled light points)
    - k - sample number being calculated
    - ω_o - direction the light is going towards (the viewer)

Area Lights: Monte Carlo Integration

$$L_d(\mathbf{x},\omega_o) \approx L_i \frac{A}{N} \sum_{k=1}^{N} f(\mathbf{x},\omega_i(k),\omega_o)\, G(\mathbf{x},\mathbf{x'}_k)\, V(\mathbf{x},\mathbf{x'}_k)$$

- G(x,x') → geometric term, accounts for how light weakens over distance
  - x' is the point sampled on the light source and typically includes:
    - inverse square falloff ($1/\text{distance}^2$)
    - cosine of angle at the surface ($n \cdot \omega_i$)
    - cosine of angle at the light source ($n' \cdot -\omega_i$)
- V(x,x') → visibility term
  - returns 1 if there's a clear line of sight between x and x'
  - returns 0 if something blocks the light
- Each x'_k represents a randomly chosen point on our light source for sample k.

Dual-Layer Light Source

- 2022 SIGGRAPH paper proposed an even fancier light source
- Casts a 2D image as a light pattern onto a 3D object
- Uses two layers:
  - Standard light source
  - “Gel” layer which modifies light color/intensity

Dual-Layer Light Source

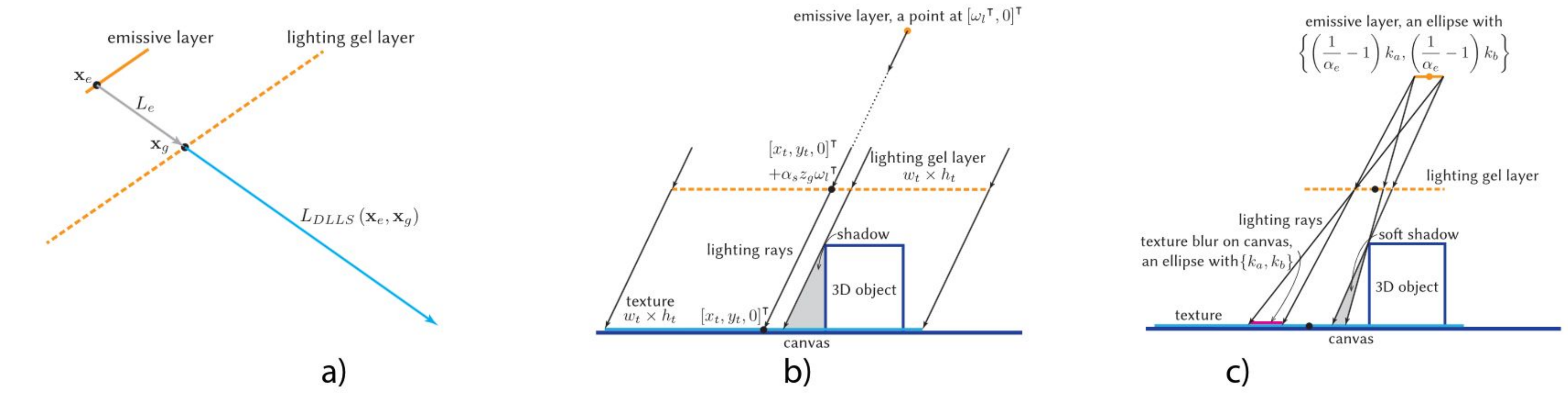


Figure 2: Scene configuration with DLSS (a) The emission of layer L_e is modulated by the lighting gel layer, providing the emission of the DLLS. (b) Shadow casting by changing the direction of lighting. (c) Anisotropic blur of lighting effects.

- Light source (emissive layer) can be a point light or an area light

Dual-Layer Light Source

A 3D model (left-top) illuminated by a 2D gel texture (left-bottom) using Dual Layer Light Source representation.

Questions?

References

- https://www.rose-hulman.edu/class/cs/csse351-abet/m4/Lighting/
- https://math.hws.edu/graphicsbook/c7/s2.html
- Fundamentals of Computer Graphics by Marschner and Shirley. Chapter 5.
- https://dl.acm.org/doi/10.1145/3532719.3543227
- https://onlinelibrary.wiley.com/doi/full/10.1111/cgf.14866