Prof. Erickson R. Nascimento
erickson@dcc.ufmg.br

LIGHT SOURCES

Computer Vision and Robotics

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

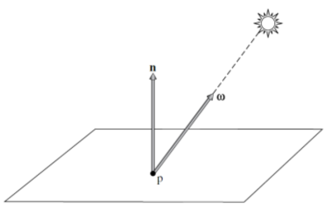
UFMG

Light sources

- For objects in a scene to be visible, some of them must emit light
- Light sources emit light, rather than scattering or absorbing

Light sources

- A Light source has:
  - Direction
  - Amount of illumination it emits



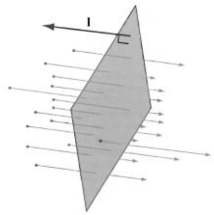
Light sources types

- We will discuss three types:
  - Directional Light
  - Omin Light
  - SpotLights

Directional Light

VERLab Computer Vision and Robotics

- Light travels in a single direction that is the same throughout the scene
  - Modeling distant light sources (e.g., sun)

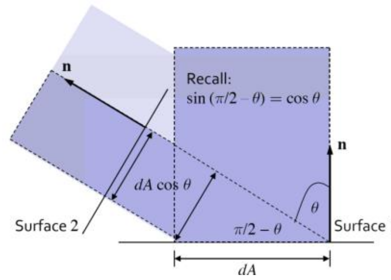


UFMG

Directional Light

VERLab Computer Vision and Robotics

- $E = E_L \cos\theta_i$
  - E_L is the irradiance perpendicular to light direction



UFMG

Directional Light

VERLab Computer Vision and Robotics

- Remember that irradiance is **additive**.
- Total irradiance from multiple directional light sources is:

$$E = \sum_{i=1}^{n} E_{Lk} \cos\theta_{ik}$$

UFMG

Point Lights

VERLab Computer Vision and Robotics

- Point lights are defined by:
  - Position p_L
  - Intensity I_L
- I_L can vary as a function of direction
- When I_L is constant the point light is called **Omni Light**

UFMG

Irradiance Revisited

- Irradiance is defined as:

$$E = \frac{d\Phi}{dA}$$

  - where Φ is the radiance flux **arriving** at the point and dA the differential area surronding the point

UFMG

Irradiance Revisited

- Irradiance for a sphere that has radius r is equals
  - Total area of a sphere: $4\pi r^2$

$$E = \frac{\Phi}{4\pi r^2}$$

  - The amount of energy received from a light falls off with the squared distance from the light

UFMG

Point Lights

- Shading equations compute the irradiance contribution of the light E_L as
  - $E_L = \frac{I_L}{r^2}$

UFMG

Point Lights

- Despite E_L decreasing proportionally to $\frac{1}{r^2}$ is physically correct, it is often preferable **distance falloff functions**
  - $E_L = I_L f_{dist}(r)$
- OpenGL fixed-function:
  - $f_{dist}(r) = \frac{1}{s_c + s_l r + s_q r^2}$

UFMG

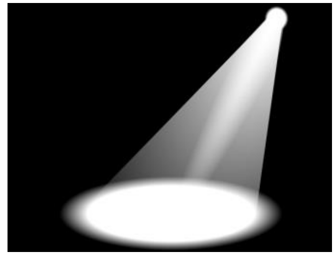
Point Lights

- **Distance falloff functions**
  - More control for lighting scene
  - Square function never reaches zero (better performance for functions that reaches zero)
  - Square function gets arbitrarily large values close to the light source

UFMG

Spotlights

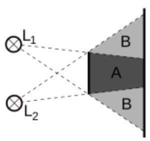
- Spotlights emit light in a cone of directions from their position

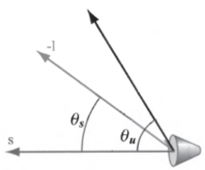


UFMG

Spotlights

- They are defined by (OpenGL fixed-fuction):
  - Falloff start: θ_s
  - Umbra angle: θ_u (A area)
  - $\boldsymbol{s}$ is the direction of the Spotlight
  - $\boldsymbol{l}$ is the direction to the surface





UFMG

Spotlights

$$I_L(\theta_s) = \begin{cases} I_{L_{max}} (cos\, \theta_s)^e, \theta_s \leq \theta_u \\ 0, \theta_s > \theta_u \end{cases}$$

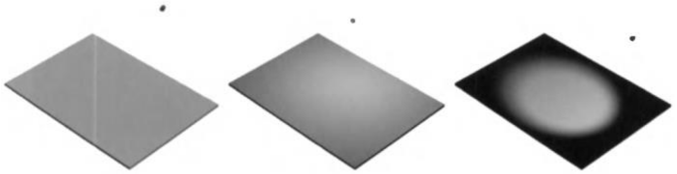
- θ_s is the angle between vector $\boldsymbol{s}$ and vector $-\boldsymbol{l}$
- The tightness of the spotlight is controlled by exponent e

UFMG

Spotlights

UFMG

Light Source Types



directional, omni with no falloff, and spotlight

UFMG