Shading

22 Feb

Diffuse shading

model "Lambertian object"

Lamberts cos Law:

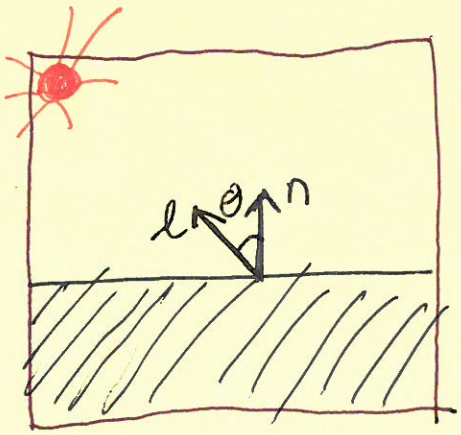
"color of a surface is proportional to the cos of the ∠ between the surface normal and a directional light source"

$$color \sim \cos\theta$$

or

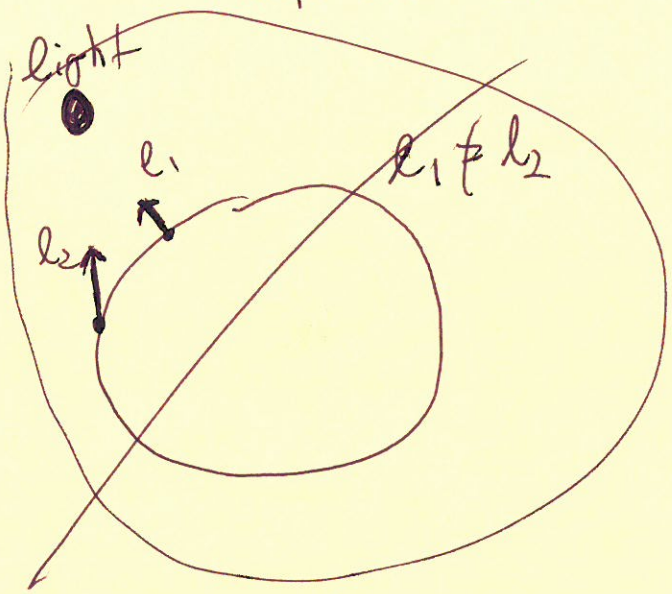
$$color \sim l \cdot n$$

we will use "C" for color



Assumption

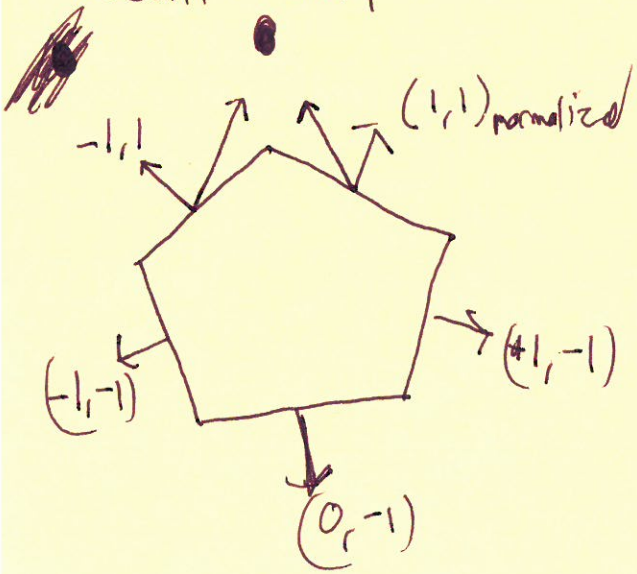
- ℓ is assumed to be independant of location on object

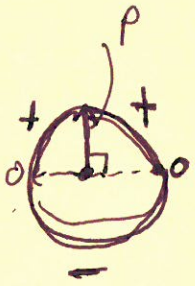


not OK

yes!

Recall dotproduct $a \cdot b = \sum_i a_i b_i$





Define diffuse reflectence c_r

$c \sim c_r \, n \cdot l$ i.e. scale color value by reflected light

Simple form of lighting w/ c_e as light intensity

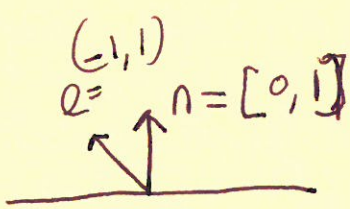
$$c = c_r c_e \, n \cdot l$$

in general $c_e \in [0,1]^3$, $c_r \in [0,1]^3$ ← rgb values

E.g.

green light $(.1, 1, 0)$

yellow surface $(.8, .7, 0)$



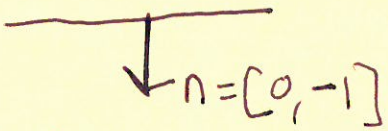
$$l_{norm} = \frac{l}{\|l\|} = \begin{bmatrix} -1/\sqrt{2} \\ +1/\sqrt{2} \end{bmatrix}$$

$$l_{norm} \cdot n = \begin{bmatrix} -1/\sqrt{2} \\ +1/\sqrt{2} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 1/\sqrt{2}$$

$c_{red} = (.1)(.8)\,1/\sqrt{2}$

$c_{green} = (1)(.7)\,1/\sqrt{2}$

$c_{blue} = (0)(0)\,1/\sqrt{2}$



to handle normals facing away from cam

$$c = c_r c_e \max(0, n \cdot l)$$

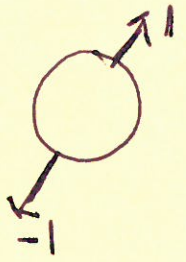
Ambient lighting

def c_a ambient light

$$c_r \left(c_a + c_e \max(0, \ell n) \right)$$

still greater than 1 so clamp

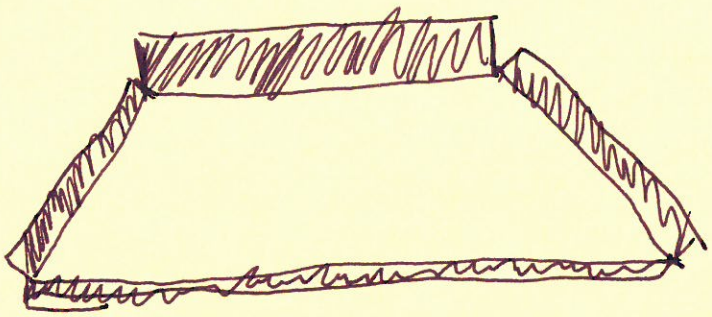
or $c_a + c_e < 1$



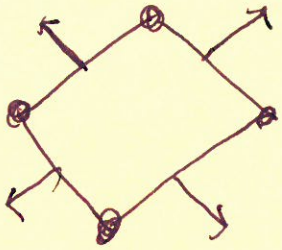
$$\frac{n \cdot l}{2} + 1$$

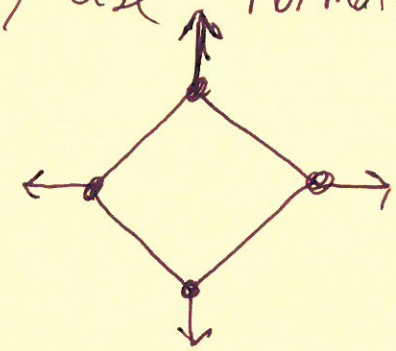
Where do we get normals

1) some models just come w/ them
2) normal of smooth surface if teselating surface
3) if given polygons . . . .
   average of normals in neighborhood



Don't use face normals, use normals at verts





NO!

it creats

facet-ing

ave of normals of Δ's in neighborhood