16. Shading Transformations

Smooth shading → surface normal for each point on sphere (implicit sphere)

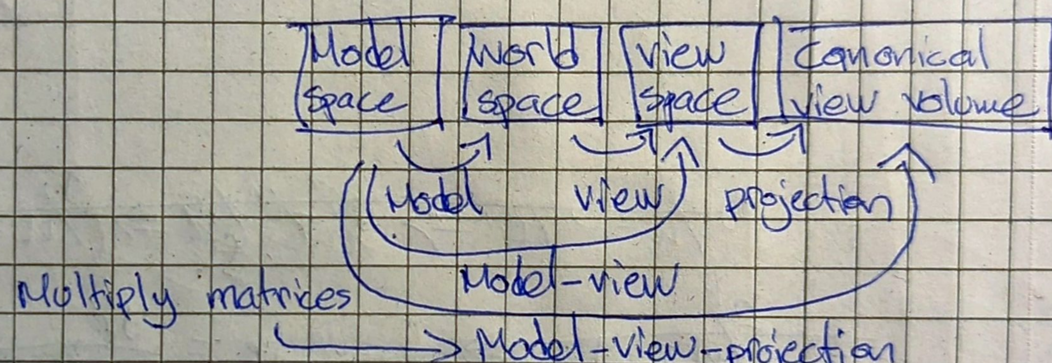
Flat shading – surface normal on each triangle

Gouraud shading – Given surface normal on each vertex, then interpolate (barycentric) color

Phong shading – Like Gouraud shading, but interpolate normals instead of colors.
NB! Normals need to be normalized.

Shading transformatinons

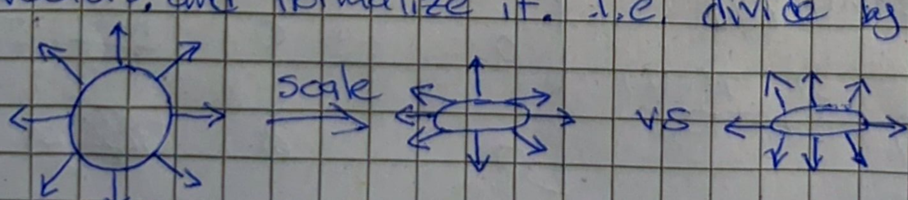
Viewing transformations – (recap) – Going from model space to world space (model transformation), then to view space (view transformation)



Shading transformations – Prefer to do in view space, not written in stone. Not canonical view volume though. Light will be defined in view space for project (stuck to camera)

(for project we should go from model to view space, then view to canonical when transforming vertices. Because we need to apply shading)

When scaling we need to take the inverse of the normal vectors, and normalize it. I.e. divide by scale.



$\mathbb{M}$ = model-view matrix

$\mathbf{p}' = \mathbb{M}\,\mathbf{p}$ ← positions
$\mathbf{n}' = (\mathbb{M}_{3\times3}^{-1})^T\,\mathbf{n}$ ← normals

17. The rendering equation

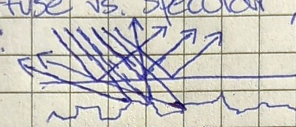
Created by Jim Kajiya, in 1986.

$$L_o(\omega_o) = \int_\Omega L_i(\omega_i) \cos\theta_i \, f_r(\omega_i, \omega_o) \, d\omega_i$$

- Ω = Hemisphere of all light dir.
- L_i = Incoming light
- ω_i = Incoming light direction
- L_o = Outgoing —"—
- ω_o = Outgoing —"—
- θ_i = Incoming light angle

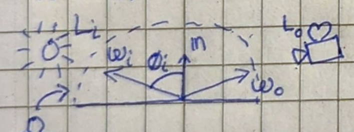
Surface reflections

~~Scattering~~ Diffuse vs. specular, can be because of coating (polish):



If some light (a beam) is specular, it needs to be subtracted from the diffuse part. (I.e. can't have 100% light on both specular and diffuse part).

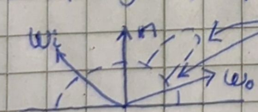
Bidirectional Reflectance Distribution Function (BRDF), $f_r(\omega_i, \omega_o)$. statistical average of amount of light scattered



$$\int_\Omega f_r(\omega_i, \omega_o)\, d\omega_o \le 1$$

$f_r(\omega_i, \omega_o) = f_r(\omega_o, \omega_i)$ ← Doesn't matter which direction we calc., how light works.

Blinn/Phong: $f_r(\omega_i, \omega_o) = K_d + K_s \dfrac{(\cos\varphi_i)^\alpha}{\cos\theta_i}$



Blinn/Phong is a crude approx. to the rendering equation.

Isotropic vs. anisotropic BRDFs

Anisotropic takes a material direction from the normal into account, making the reflection different based on the direction of the object.