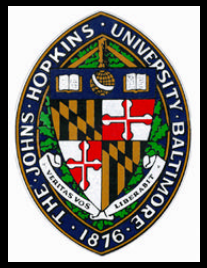
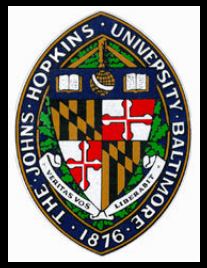


A Non-Photorealistic Lighting Model for Automatic Technical Illustration

from Gooch, Gooch, Shirley, and Cohen, *SIGGRAPH 98*.

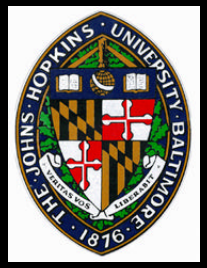


Technical Illustration Goals

Shape information more important than photorealism

"Make all visual distinctions as subtle as possible, but still clear and effective"

- **Tufte**



Technical Illustration Characteristics

Edge lines drawn with black curves

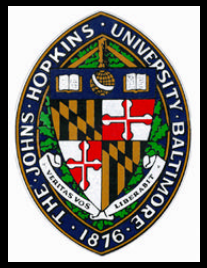
- **boundaries, silhouettes, discontinuities**

White highlights from single light source

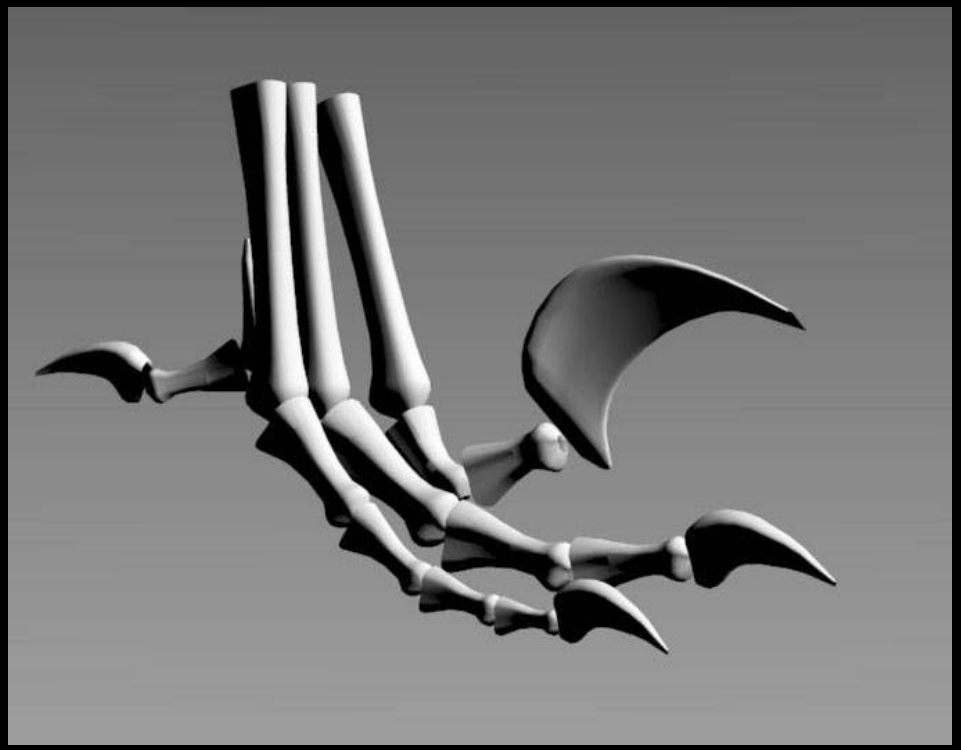
Shading stays far from black and white

- **limited intensity range**

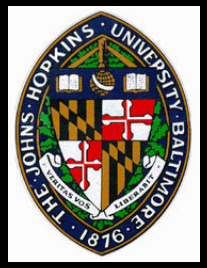
Hue changes (warm to cool) help to indicate surface normal



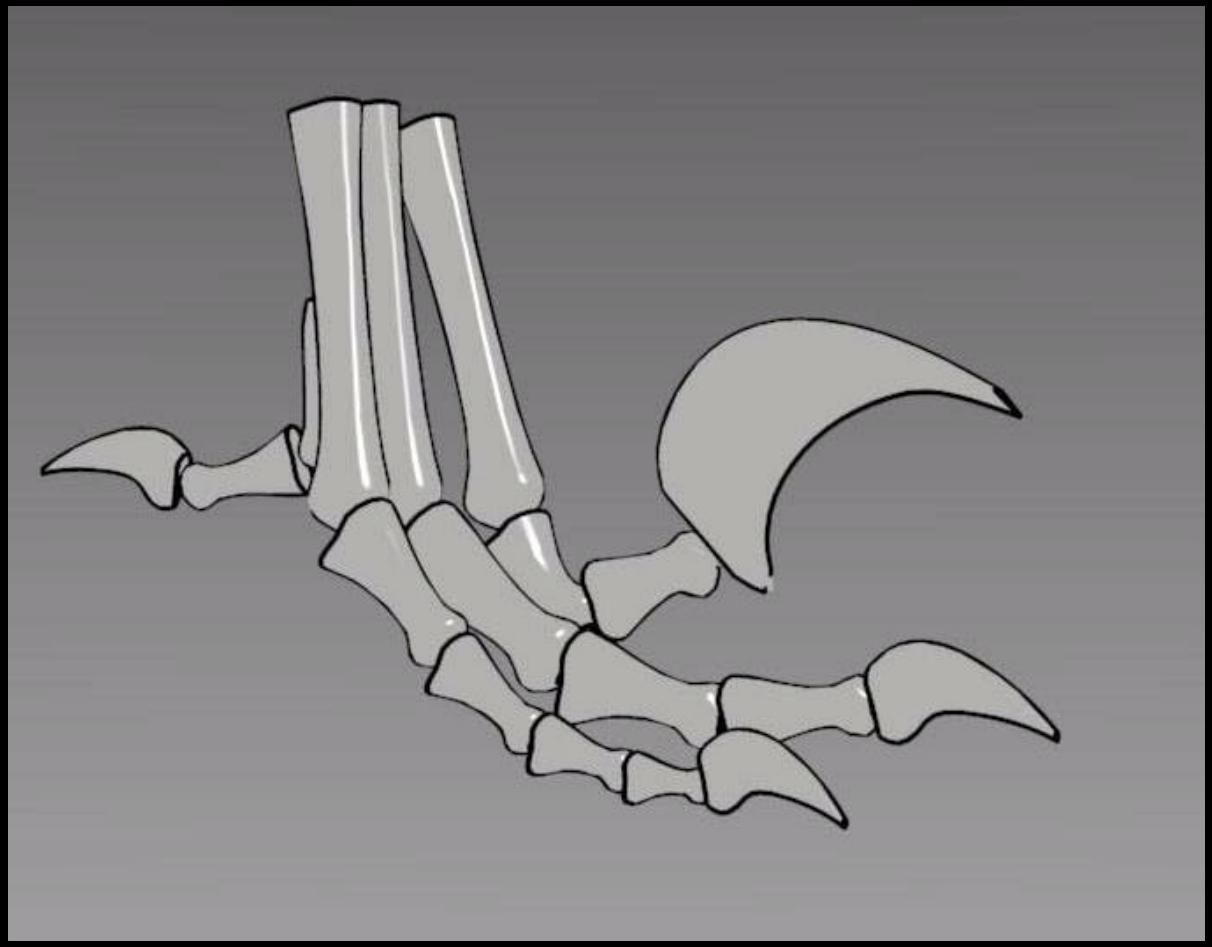
Phong Illumination

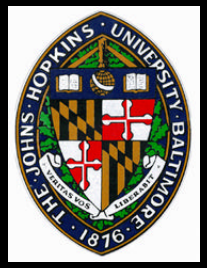


highlights lost
edge lines would be hard to see
no variation dark regions

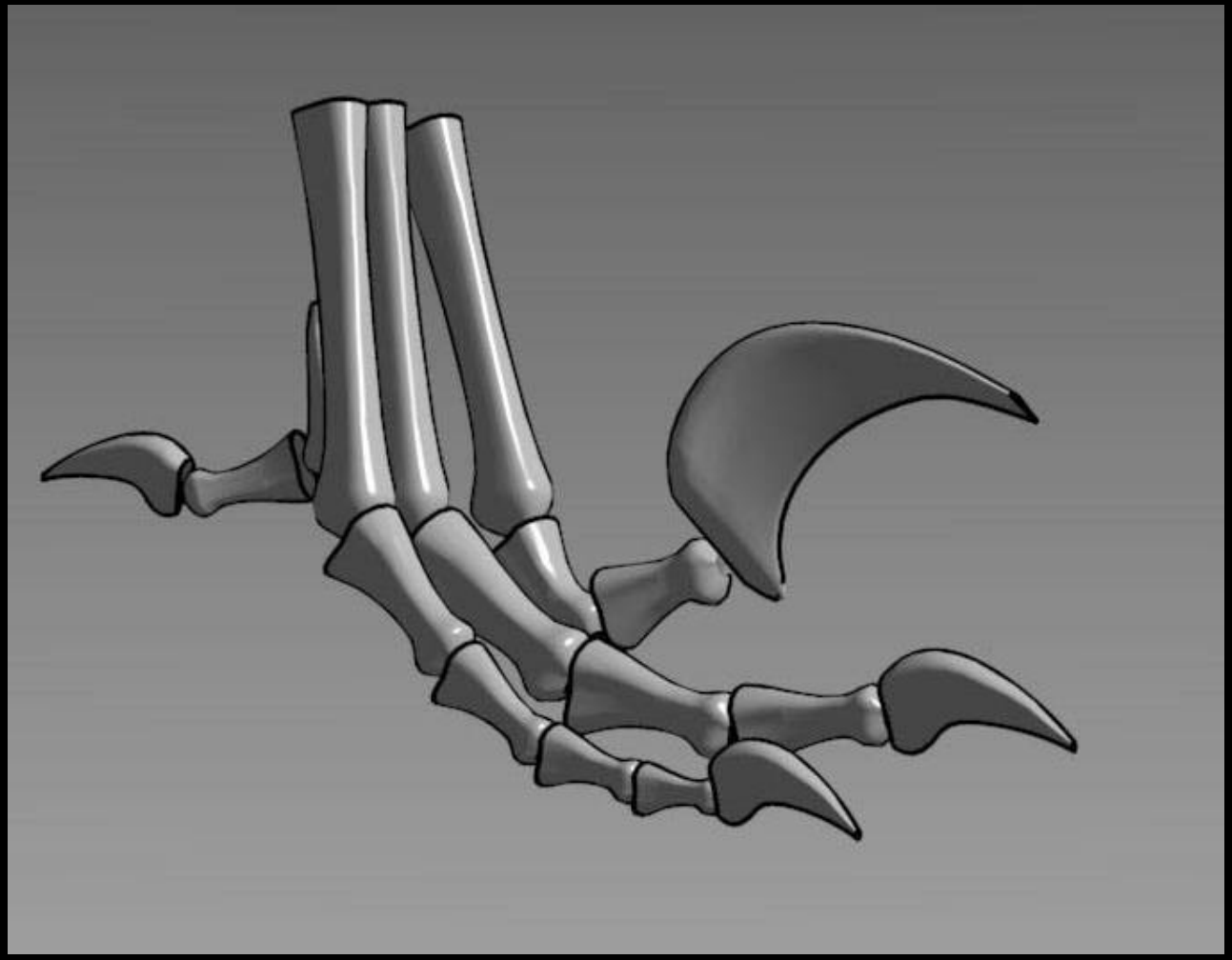


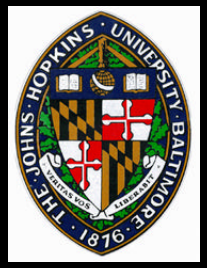
Solid Color + Highlights and Edges





Restricted Intensity Phong + Edges





Diffuse Illumination

Standard Lambertian Model

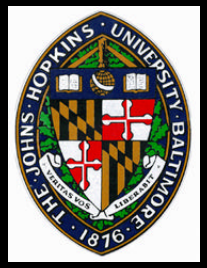
$$I = k_d * k_a + k_d * \max(0 , l.n)$$

- Points with normals away from light all constant color

Color Interpolation Model

$$I = (1+ l.n)/2 * k_1 + [1 - (1+ l.n)/2] * k_2$$

- Variation across entire range of normals
  - $l.n \in [-1,1]$



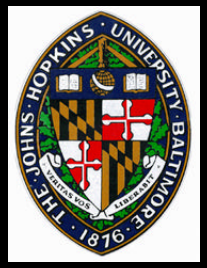
Color Temperature Principles

Warm colors approach

- **Red, yellow, orange**

Cool temperatures recede

- **Blue, violet, green**



Cool-to-Warm Illumination

Blue-to-yellow illumination

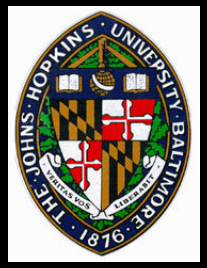
- k_1 = blue = (0,0,b)
- k_2 = yellow = (y,y,0)

Scaled object-color illumination

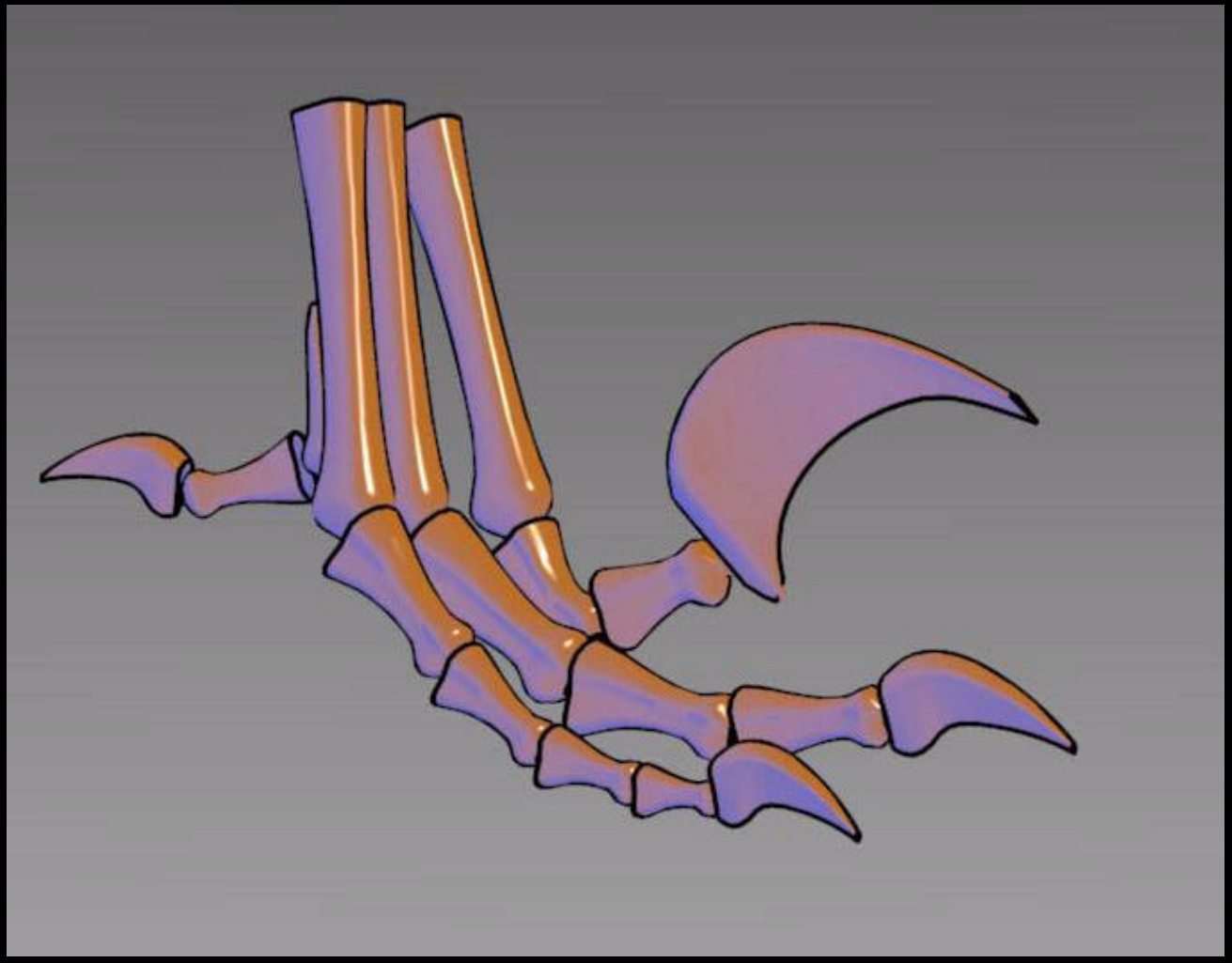
- k_1 = black = (0,0,0)
- k_2 = object color = k_d

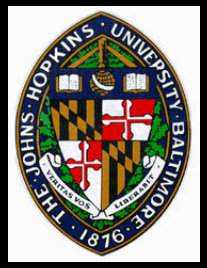
Combined model

- $k_1 = k_{cool} = (0,0,b) + \alpha k_d$
- $k_2 = k_{warm} = (y,y,0) + \beta k_d$

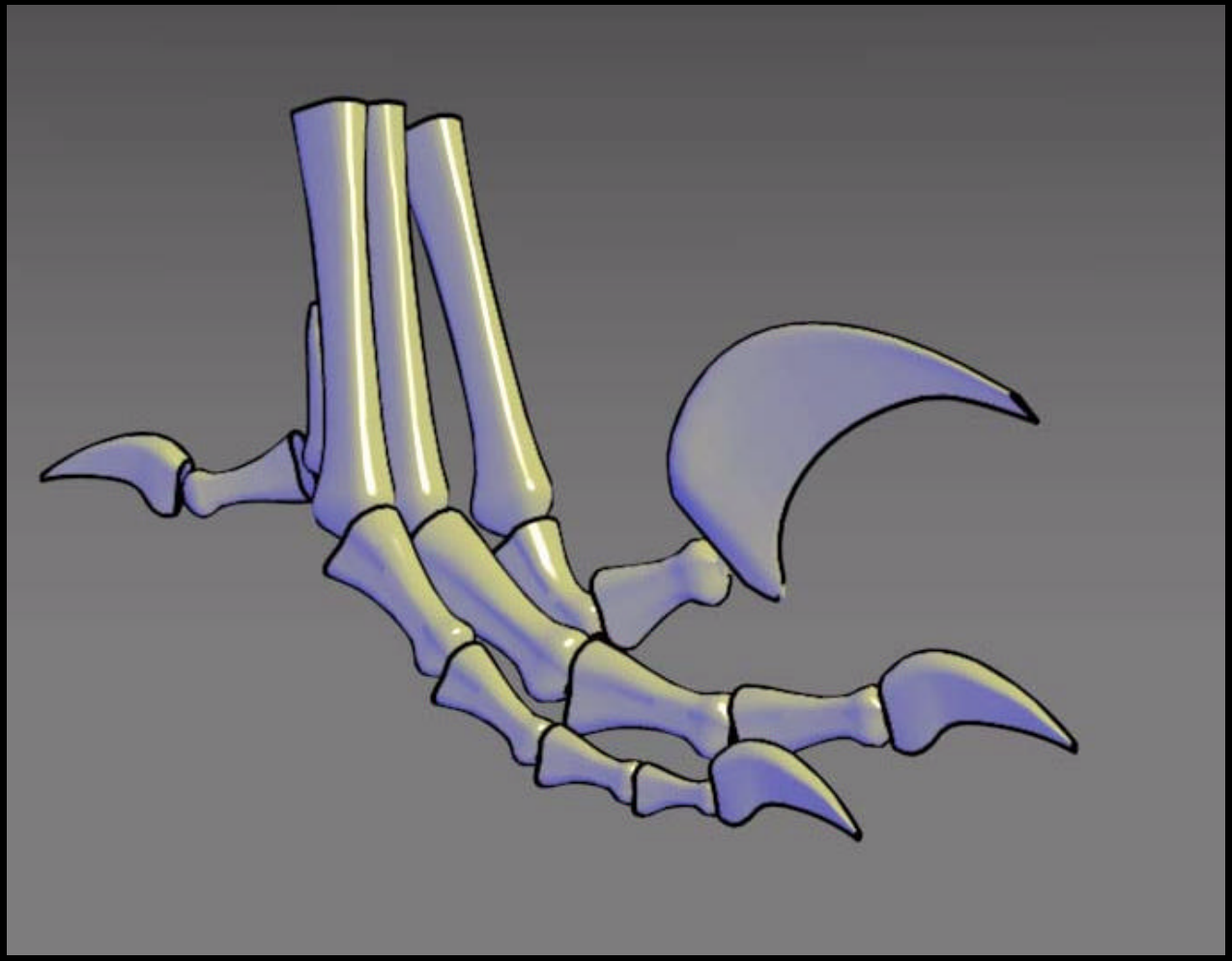


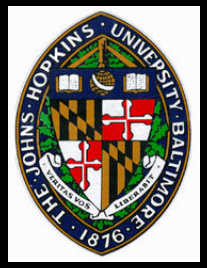
Constant Luminance, Changing Hue





Changing Hue and Luminance





Approximating Cool-to-Warm Illumination in OpenGL

Two directional lights

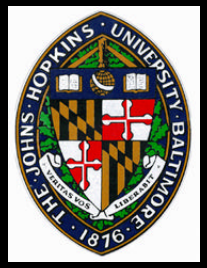
- **Direction L, intensity $(k_{warm} - k_{cool})/2$**
- **Direction -L, intensity $(k_{cool} - k_{warm})/2$**
  - **Negative intensities are legal!**

Ambient light

- **Intensity $(k_{cool} + k_{warm})/2$**

White surface color

Add white highlights using second pass

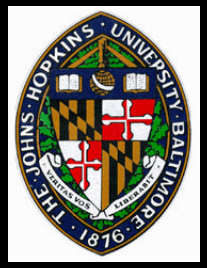


Illustrative Metal Shading

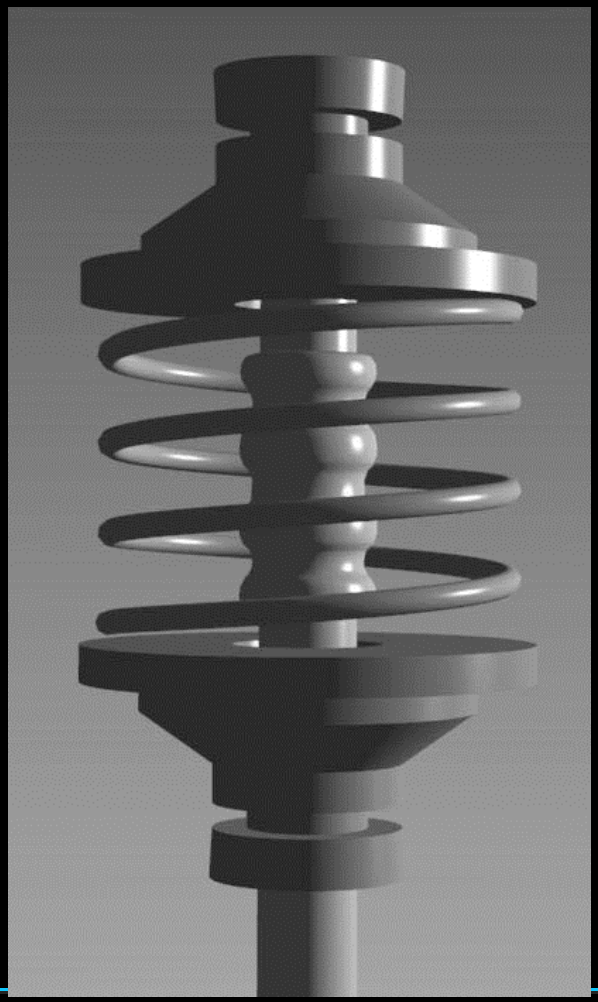
Milled metals exhibit streaks along milling axis

Simulate this anisotropy using stripes of various intensities along milling axis

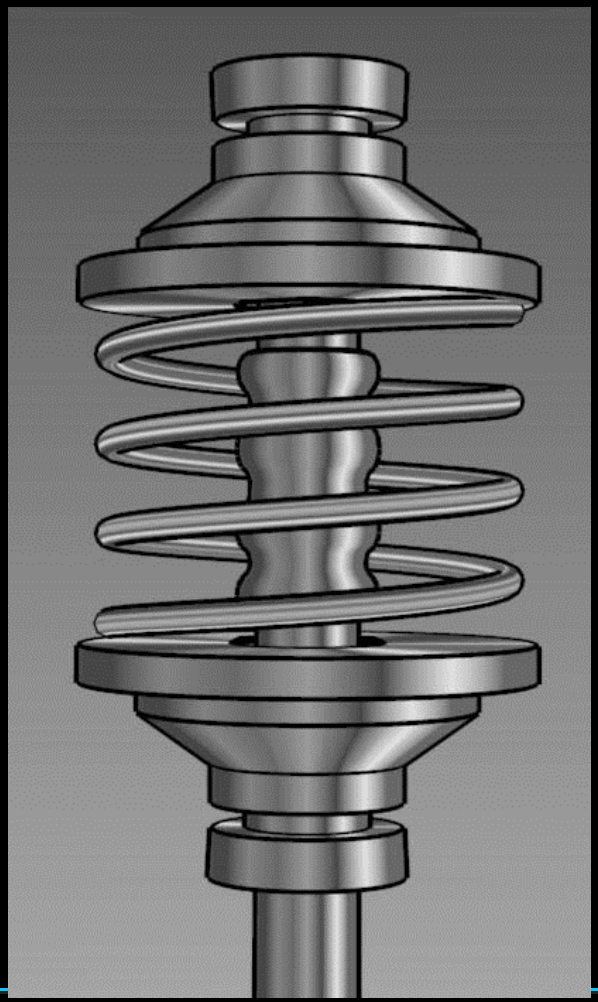
- **Random stripe intensities from 0.0 to 0.5**
- **Stripe closest to light direction is white**
- **Linearly interpolate colors between stripes**



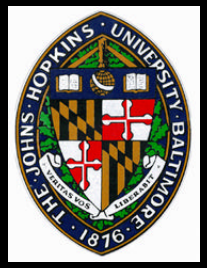
Metal Shading + Edges



Phong



Metal



Video

Gooch, Sloan, Gooch, Shirley, and Riesenfeld, "Interactive Technical Illustration," ***Proceedings of 1999 Symposium on Interactive 3D Graphics*****.**