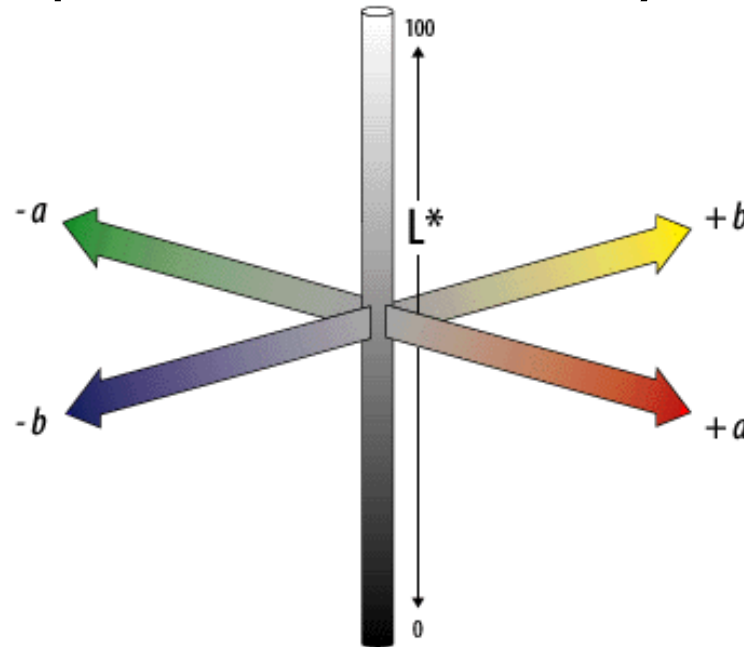


SHADERS

CIE $L^*a^*b^*$

2

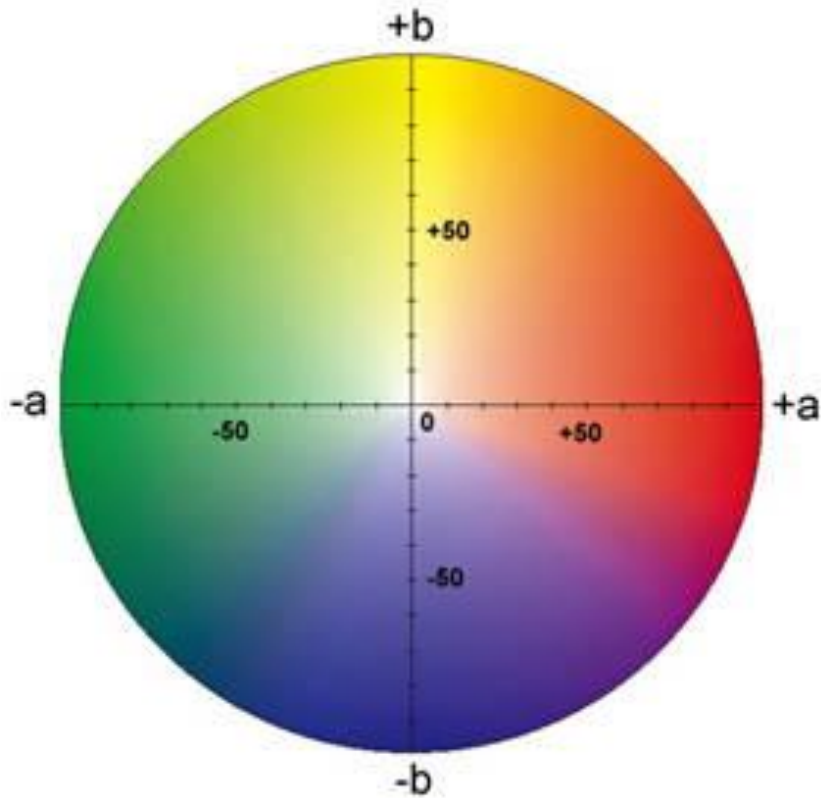
- Includes all perceivable colors
- Perceptually uniform
- L - lightness, close match to human perception
- a – color component from green to red
- b – color component from blue to yellow



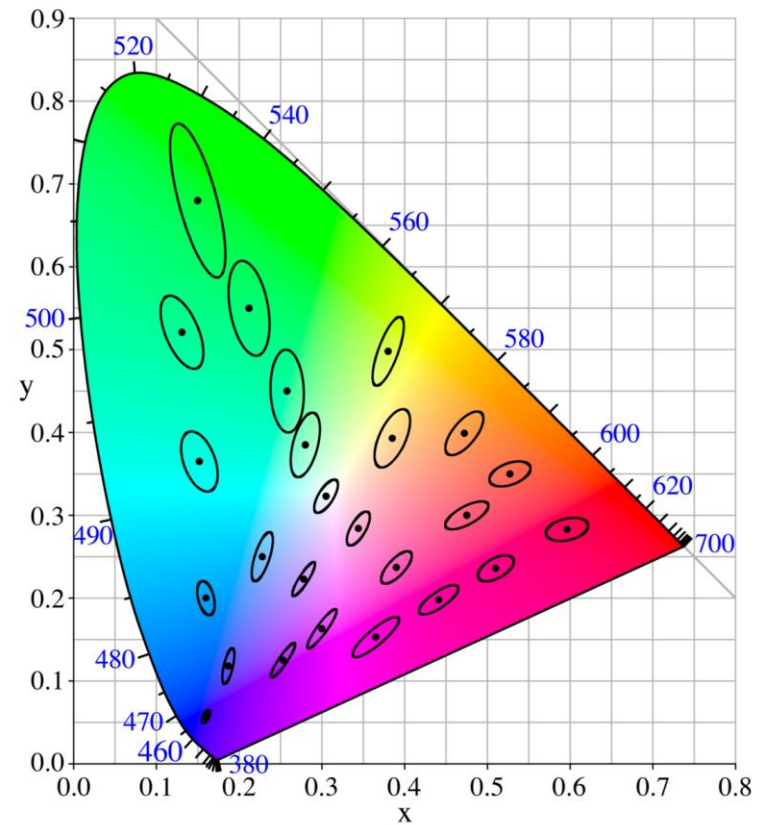
CIE Lab vs. RGB

3

CIE Lab



RGB

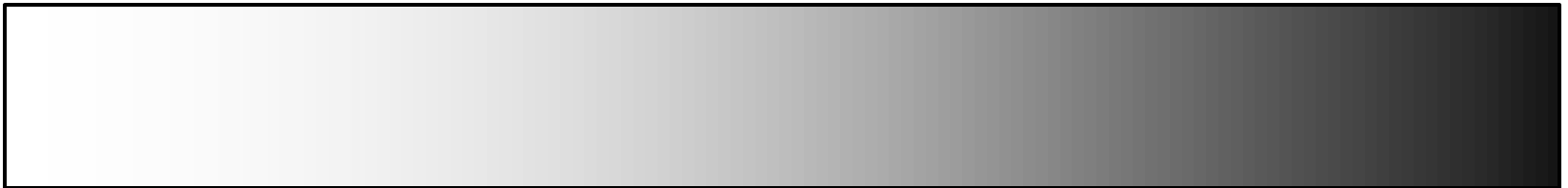




Toon Shader

5

- Discretize diffuse and specular factor
 - ~ 4 intensity values for diffuse factor
 - ~ 3 intensity values for specular factor





Cook Torrance Shader

7

- Surfaces are composed of microfacets:
 - ▣ Reflect incoming light
 - ▣ Multiple facets rendered in single pixel
 - ▣ Rough surface = slope varies greatly
 - ▣ Smooth surface = similarly oriented microfacets
- Focuses on specular reflection

$$\text{specularColor} = (\mathbf{n} \cdot \mathbf{l}) * \text{specular} * (\text{SunColor} \wedge \text{MaterialColor})$$

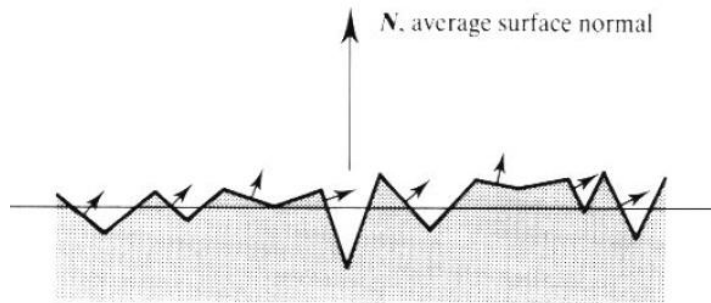
Where: $\text{specular} = \frac{F_{\lambda}(\theta) * D * G}{\pi(\mathbf{n} \cdot \mathbf{l})(\mathbf{n} \cdot \mathbf{v})}$

$F_{\lambda}(\theta)$ Fresnel
 D distribution of microfacets
 G geometric attenuation

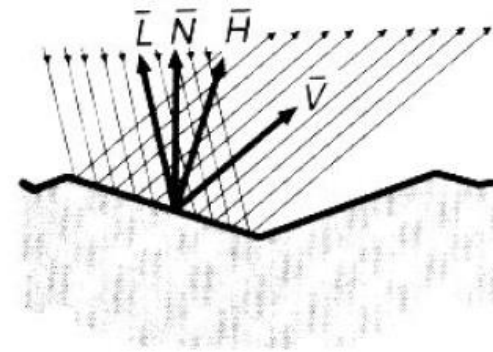
Microfacet Matovation

8

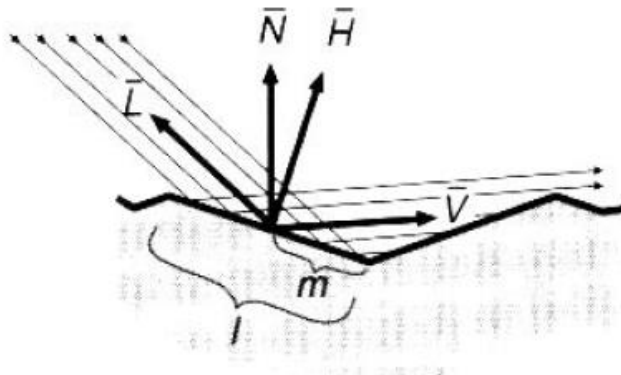
Surface composed by microfacets:



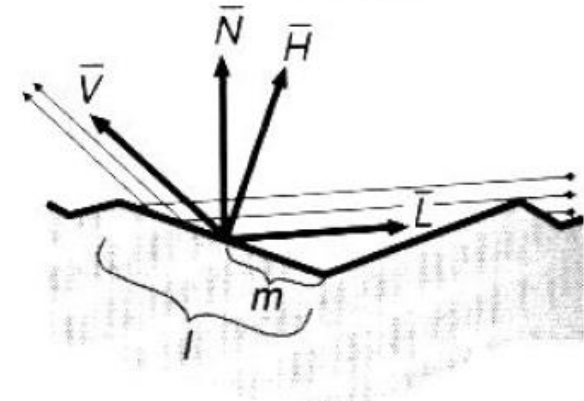
Illuminated microfacet:



Masking of reflected light:



Shadowing of incoming light:



Geometric Attenuation

9

- Microfacets block incoming light
- Value from $[0, 1]$ which represents remaining light
- Microfacets are assumed to be V-shaped grooves
- There are three cases, final factor is minimal value

The light is reflected without interference: $G_a = 1$

Light is blocked after reflection: $G_b = \frac{2(\mathbf{n} \cdot \mathbf{h})(\mathbf{n} \cdot \mathbf{v})}{\mathbf{v} \cdot \mathbf{h}}$

Light is blocked before reaching next microfacet: $G_c = \frac{2(\mathbf{n} \cdot \mathbf{h})(\mathbf{n} \cdot \mathbf{l})}{\mathbf{l} \cdot \mathbf{h}}$

Final attenuation factor: $G = \min(G_a, G_b, G_c)$

Roughness – Beckmann distribution

10

- Defines fraction of microfacets oriented the same way as half vector **h**
 - ▣ On smooth surfaces all light is close to specular reflection
 - ▣ On rough surfaces the light is more distributed
- Can be calculated with e.g. Beckmanns distribution

$$D = \frac{1}{\pi m^2 \cos^4 \alpha} e^{-\left(\frac{\tan \alpha}{m}\right)^2} = \frac{1}{\pi m^2 \cos^4 \alpha} e^{\left(\frac{(\mathbf{n} \cdot \mathbf{h})^2 - 1}{m^2 (\mathbf{n} \cdot \mathbf{h})^2}\right)}$$

Where: m is material roughness

Fresnel – Schlick approximation

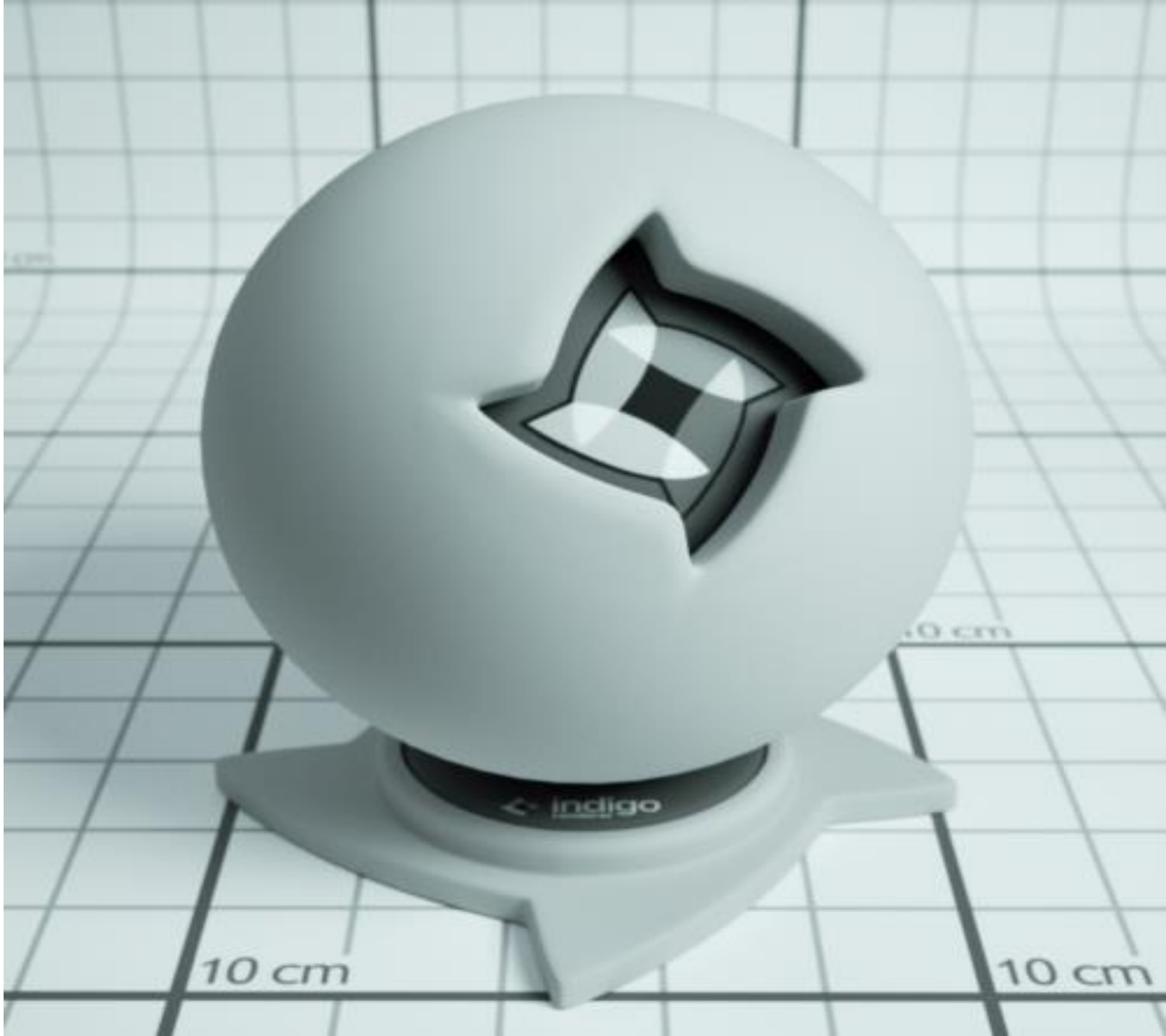
11

- Defines what fraction of incoming light is reflect and transmitted
- Schlick approximation is used, due to complexity of original formula

$$F_{\lambda}(\theta) = f_{\lambda} + (1 - f_{\lambda})(1 - \theta)^5$$

Where: f_{λ} reflectance at normal distance

$\theta = \mathbf{h} \cdot \mathbf{v}$ angle between half and view vectors



Oren Nayar Shader

13

- Lambertian model inappropriate for many materials
- Surfaces can be modeled by microfacets
- Camera projects several facets into one pixel
- Takes into account masking, shadowing, interreflections
- Takes a single parameter the roughness of a surface
- More info in original paper:
 - ▣ http://www1.cs.columbia.edu/CAVE/publications/pdfs/Oren_SIGGRAPH94.pdf

Oren Nayar Shader - Formulas

14

\mathbf{n} = normal \mathbf{l} = light direction \mathbf{v} = view direction \mathbf{e} = eye direction

$$\alpha = \max(\angle \mathbf{nv}, \angle \mathbf{nl})$$

$$\beta = \min(\angle \mathbf{nv}, \angle \mathbf{nl})$$

$$A = 1 - 0.5 \frac{\text{roughness}^2}{\text{roughness}^2 + 0.57}$$

$$B = 0.45 \frac{\text{roughness}^2}{\text{roughness}^2 + 0.09}$$

$$C = \sin \alpha * \tan \beta$$

$$\gamma = (\mathbf{e} - \mathbf{n}(\mathbf{e} \cdot \mathbf{n})) \cdot (\mathbf{l} - \mathbf{n}(\mathbf{l} \cdot \mathbf{n}))$$

$$L_1 = \max(0, \mathbf{n} \cdot \mathbf{l}) * (A + B * \max(0, \gamma) * C)$$

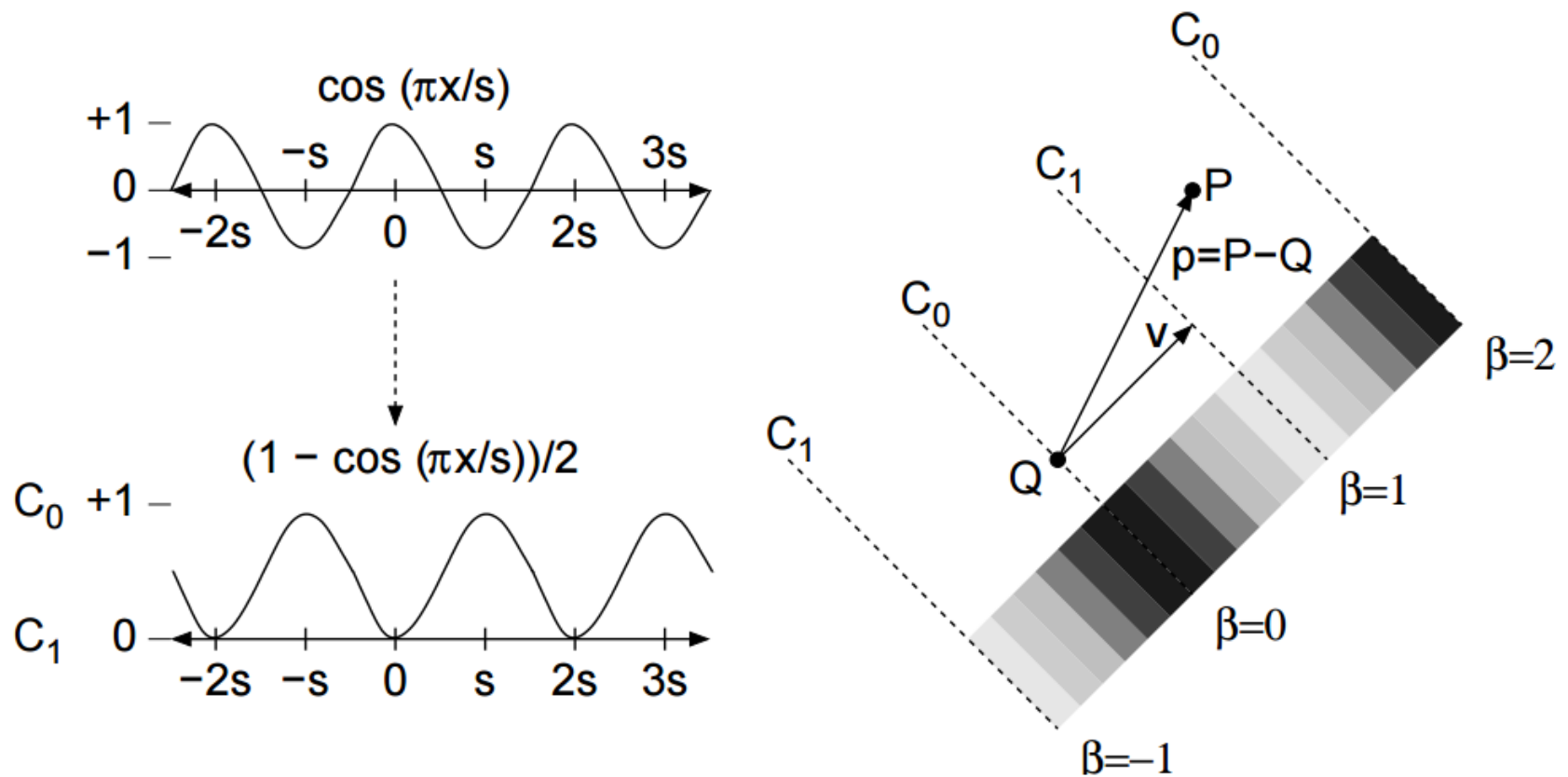
Gradient Shader (1)

15

- Creates cosinusoidal wave
- Project vector from origin to point onto gradient direction
- Calculate cosine of gradient value
- Transform cosine from $[-1, 1]$ to $[0, 1]$ to get alpha
- Use alpha blending between two shaders S_0 and S_1

Gradient Shader (2)

16



17

Questions?