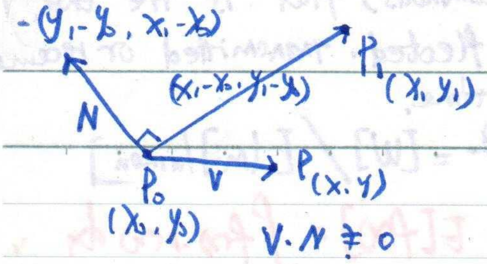
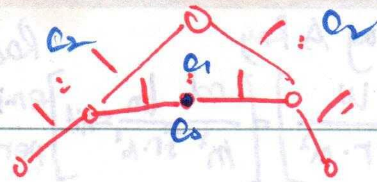


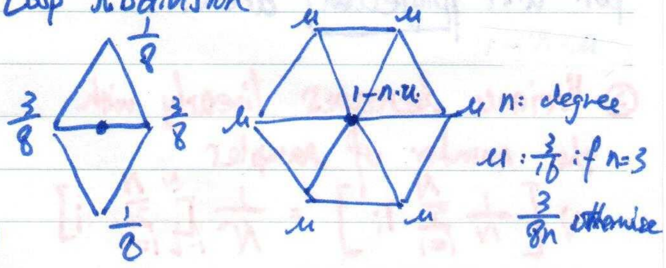
$V \cdot N \neq 0$



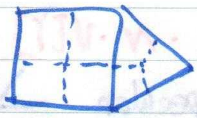
Bézier Curve : General Algebraic Formula

$$\begin{cases} b^n(t) = \sum_{j=0}^{n} b_j B_j^n(t) \\ B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i} \ \text{(Bernstein polynomials)} \end{cases}$$

Loop Subdivision



Catmull-Clark



Transformation $V' = T \cdot V$

① Scale $\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$

② Rotation (ccw) $\begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$

③ Translation $\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$

Shear-x ④ $\begin{bmatrix} 1 & S_x \\ 0 & 1 \end{bmatrix}$

$$\begin{cases} x' = x + S_x y \\ y' = y \end{cases}$$

Möller Trumbore Algorithm

$$\vec{O} + t\vec{D} = (1-b_1-b_2)\vec{P_0} + b_1\vec{P_1} + b_2\vec{P_2}$$

$$\begin{bmatrix} t \\ b_1 \\ b_2 \end{bmatrix} = \frac{1}{\vec{S_1}\cdot\vec{E_1}} \begin{bmatrix} \vec{S_2}\cdot\vec{E_2} \\ \vec{S_1}\cdot\vec{S} \\ \vec{S_2}\cdot\vec{D} \end{bmatrix}$$

$\vec{E_1} = \vec{P_1} - \vec{P_0}$

$\vec{E_2} = \vec{P_2} - \vec{P_0}$

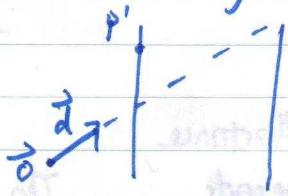
$\vec{S} = \vec{O} - \vec{P_0}$ $\qquad \vec{a} \times \vec{b} = \begin{vmatrix} i & j & k \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix}$

$\vec{S_1} = \vec{D} \times \vec{E_2}$

$\vec{S_2} = \vec{S} \times \vec{E_1}$

Optimize Ray-Plane Intersection For Axis-Aligned Planes

$$t = \frac{(p' - o) \cdot N}{d \cdot N} \ \text{(general)}$$



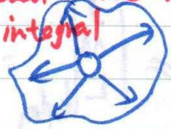
$$t = \frac{p'_x - o_x}{d_x}$$

Snell's Law

$$\eta_i \sin\theta_i = \eta_t \sin\theta_t$$

Russian Roulette

Radiant Intensity: the power per unit solid angle emitted by a point light source



$$\left[\frac{W}{sr}\right] = \left[\frac{lm}{sr} = cd = candela\right]$$

$$\left[\frac{W}{m^2}\right] = \left[\frac{lm}{m^2} = lux\right]$$

Irradiance: the power per unit area incident on a surface point

Solid Angles: $\Omega = \frac{A}{r^2}$ | Sphere has 4π steradians

A randomized integral estimator is unbiased if its expected value is the desired integral

Monte Carlo Integration

$$F_N = \frac{1}{N}\sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)} \ \text{(general)}$$

$$F_N = \frac{(b-a)}{N}\sum_{i=1}^{N} f(X_i) \ \text{(basic)}$$

$\begin{cases} \text{Importance sampling see ①} \\ \text{Hemisphere sampling} \end{cases}$

Light traveling Along A Ray

$$\left[\frac{W}{sr \cdot m^2}\right] \left[\frac{cd}{m^2} = \frac{lm}{sr \cdot m^2} = nit\right]$$

Radiant (luminous) flux is the energy emitted, reflected, transmitted or received per unit time.

$$\Phi = \frac{dQ}{dt} = [W] / [lm = lumen]$$

Radiance: fundamental field quantity that describes the distribution of light in an environment
The power emitted, reflected, transmitted or received by a surface, per unit solid angle per unit $\boxed{\text{projected}}$ area.

① $E[Y] = E[f(X)] = \int f(x) \cdot p(x)\, dx$

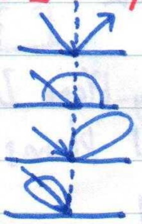
$$E[F_N] = E\left[\frac{1}{N}\sum_{i=1}^{N}\frac{f(X_i)}{p(X_i)}\right]$$
$$= \frac{1}{N}\sum_{i=1}^{N} E\left[\frac{f(X_i)}{p(X_i)}\right]$$
$$= \frac{1}{N}\sum_{i=1}^{N}\int_a^b \frac{f(x)}{p(x)} \cdot p(x)\, dx$$
$$= \frac{1}{N}\sum_{i=1}^{N}\int_a^b f(x)\, dx$$
$$= \int_a^b f(x)\, dx$$

② Variance decreases linearly with the number of samples

$$V\left[\frac{1}{N}\sum_{i=1}^{N} Y_i\right] = \frac{1}{N^2}\sum_{i=1}^{N} V[Y_i]$$
$$= \frac{1}{N^2} \cdot N \cdot V[Y] = \frac{1}{N} V[Y]$$

全反射：从光密到光疏

Ideal specular
Ideal diffuse
Glossy specular
Retro-reflective



$$\frac{1}{\text{object distance}} + \frac{1}{\text{image distance}} = \frac{1}{\text{focal length}}$$

$$\frac{1}{z_o} + \frac{1}{z_i} = \frac{1}{f}$$

BRDF: The bidirectional reflectance distribution function represents how much light is reflected into each outgoing direction w_r from each incoming direction w_i

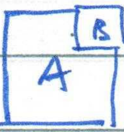
The aperture plane resolution is determined by the number of pixels under each microlens
The sensor plane resolution is determined by the number of microlenses

F-number: $\frac{\text{focal length}}{\text{diameter of aperture}}$
FoV: Field of View
DoF: Depth of Field.

$$\begin{bmatrix} S \\ M \\ L \end{bmatrix}_{display} = \begin{bmatrix} — r_S — \\ — r_M — \\ — r_L — \end{bmatrix} \begin{bmatrix} | \\ s_{disp} \\ | \end{bmatrix}$$
$$= \begin{bmatrix} — r_S — \\ — r_M — \\ — r_L — \end{bmatrix} \begin{bmatrix} | & | & | \\ s_R & s_G & s_B \\ | & | & | \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} ? \Leftarrow$$

$$\begin{bmatrix} S \\ M \\ L \end{bmatrix}_{real} = \begin{bmatrix} — r_S — \\ — r_M — \\ — r_L — \end{bmatrix} \begin{bmatrix} | \\ s \\ | \end{bmatrix}$$



Moiré Pattern
Jaggies
False Motion

before sampling

Pixel Fill Factor $= \frac{A}{A+B}$ Antialiasing: filter out High frequency

it will increase while pixel size increasing

Nyquist frequency: half of sampling frequency

Mipmap computation $\begin{cases} L = \max\left(\sqrt{\left(\frac{du}{dx}\right)^2 + \left(\frac{dv}{dx}\right)^2}, \sqrt{\left(\frac{du}{dy}\right)^2 + \left(\frac{dv}{dy}\right)^2}\right) \\ D = \log_2 L \end{cases}$

$\begin{cases} \text{Diffuse shading} & L_d = k_d \left(\frac{I}{r^2}\right) \max(0, n \cdot l) \\ \text{Specular shading} & L_s = k_s \left(\frac{I}{r^2}\right) \max(0, n \cdot h)^p \\ \text{Ambient shading} & L_a = k_a I_a \end{cases}$

Blinn-Phong: $L = L_a + L_d + L_s$

Forward kinematics: Provide angles, computer determines final position

Inverse kinematics: Provide position, computer compute the joint angles (multiple solutions or no solutions) (may not realistic enough)

Combat Instability

- Modified Euler
- Adaptive step size
- implicit method (unconditionally stable)
- Position-based/Verlet Integration

Metamers: two different spectra (∞-dim) that project to the same (S, M, L) response

CIE chromaticity diagram: Pure colors are at the edge of the plot and become more desaturated as you go towards the centroid of the plot. (luminance values are dealt with separately - No black)

CIELAB: perceptual uniformity across colors. L^* is lightness, a^* red-green, b^* blue-yellow ($L^*a^*b^*$)

HSV, CIELAB, YCbCr are perceptual organized color space

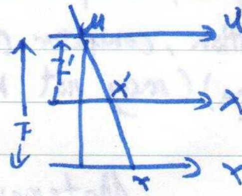
Signal-to-Noise Ratio (SNR) $= \frac{\mu}{\sigma}$

$SNR(dB) = 20 \log_{10}\left(\frac{\mu}{\sigma}\right)$

In Forward Euler, we estimate the velocity and position of each particle for the next time step on the acceleration and velocity of the particle at the current time step. Simple but unstable

Verlet integration uses relative low stiffness and modified Euler steps (using the derivative at the end of the step) to stabilize the forward step. Then, it applies constraints iteratively on the position of the particles to prevent the spring from being too stretched out. Finally, the velocity for each particle are calculated using the difference between the final, constrained positions and their positions at the previous time step.

We may choose to use it because it gives a more stable simulation than Forward Euler. In Verlet, the constraints and method of estimating the velocity effectively throw away energy and stabilize the simulation.



$L_F'(x', y', u, v)$

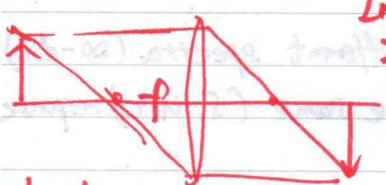
$X = L_F\left(u + \frac{x'-u}{\alpha}, v + \frac{y'-v}{\alpha}, u, v\right)$

where $\alpha = \frac{F'}{F}$

$E_{F'} = \frac{1}{\alpha^2 F^2} \iint X \, du \, dv$

unit magnification

In this case $z_o = z_i$



object z_o

image z_i

$\frac{1}{f} = \frac{1}{z_o} + \frac{1}{z_i}$

Compute the fraction of light reflected from a smooth dielectric surface: Fresnel equation

Displacement mapping actually changes the surface, while bump mapping only changes the normal. So they'll give different silhouettes to the same object.

metamers

CMY: subtractive model.

gamut

Incoming light rays meant for one pixel are accidentally detected by a different, nearby pixel

$\Rightarrow$ Optical cross-talk

$\left\{\begin{array}{l}\text{structural}\\ \text{shearing}\\ \text{bending}\end{array}\right.$