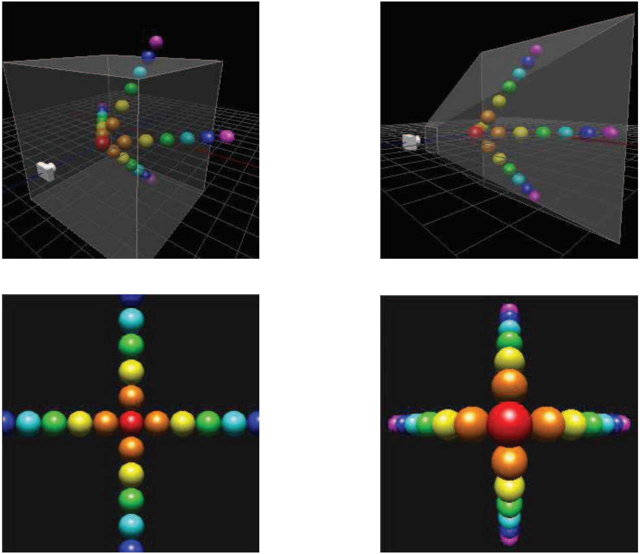
Introduction to Computer Science
Lecture 9: Computer 3D Graphics

Tian-Li Yu

Taiwan Evolutionary Intelligence Laboratory (TEIL)
Department of Electrical Engineering
National Taiwan University
tianliyu@cc.ee.ntu.edu.tw

Slides made by Tian-Li Yu, Jie-Wei Wu, and Chu-Yu Hsu

Parallel vs. Prospective Projection



Simple Projection

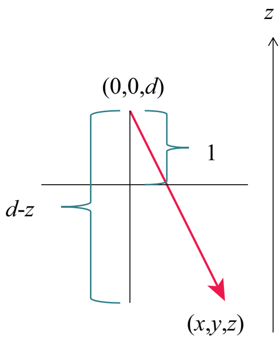
- Suppose camera is fixed at $(0, 0, d)$.
- Projection plane is fixed at $z = d - 1$
- Usually keep z information

- Parallel projection

$$(x, y, z) \rightarrow (x, y, z)$$

- Prospective projection

$$(x, y, z) \rightarrow (\frac{x}{d-z}, \frac{y}{d-z}, z)$$



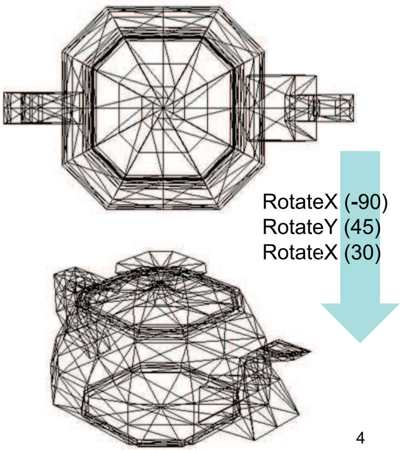
Translations & Rotations

- Translations
  - Simply add a vector.
- Rotations

$$R_X(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$

$$R_Y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$R_Z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

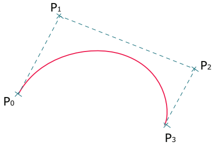


Modeling

- Typically use triangles and quadrangles.
- Easy to compute the normal vector.
- How to increase resolution?
  - Spline and Bézier curves

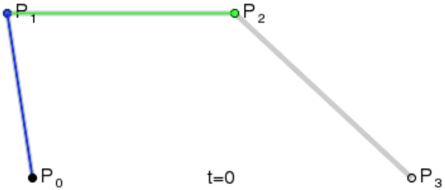
Bézier Curves

- May be defined on different degree of polynomials.
- Here we introduce the cubic one with 4 control points:



$$((1-t)+t)^3 = (1-t)^3 + 3t(1-t)^2 + 3t^2(1-t) + t^3$$

$$P(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t) P_2 + t^3 P_3$$



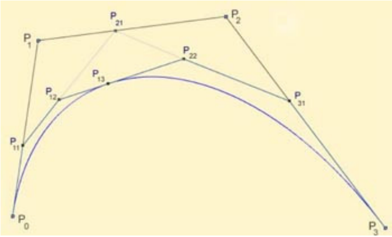
Recursive Subdivision

- A cubic Bézier curve can be subdivided into two cubic curves.

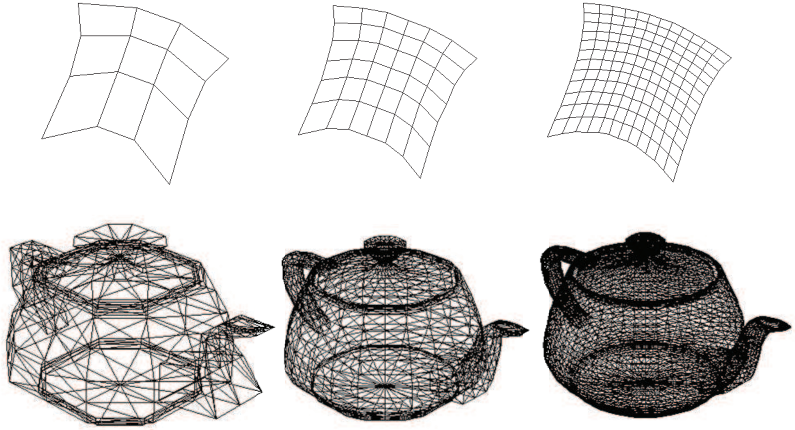
  Given any t (0.4),

  - $P_{11} = (1-t)P_0 + tP_1$
  - $P_{21} = (1-t)P_1 + tP_2$
  - $P_{31} = (1-t)P_2 + tP_3$
  - $P_{12} = (1-t)P_{11} + tP_{21}$
  - $P_{22} = (1-t)P_{21} + tP_{31}$
  - $P_{13} = (1-t)P_{12} + tP_{22}$

  - $P_0, P_{11}, P_{12}, P_{13}$ forms one curve.
  - $P_{13}, P_{22}, P_{31}, P_3$ forms the other.



Recursive Subdivision

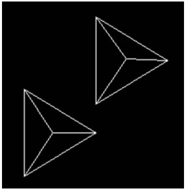


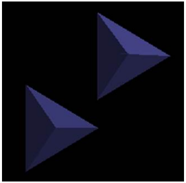
Lighting

- Ambient lighting
- Directional lighting
- Point lighting

$$I = I_a + \frac{(1 - I_a)}{1 + \alpha d + \beta d^2}$$

- d: distance between light source and target.
- I_a: intensity of ambient light.

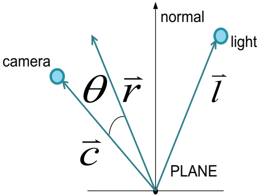




Shading

$\vec{n}$: normal vector of plane.
$\vec{l}$: normal vector of light.
$\vec{c}$: normal vector of camera.
$\vec{r}$: normal vector of reflection.

- Phong model
  - Ambient + Diffusion + Specular
- Diffusion: $\vec{n} \cdot \vec{l}$
- Specular: $(\cos\theta)^s$
  - Higher s → more mirror-like surface

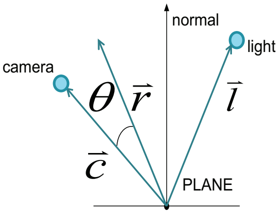


Specular

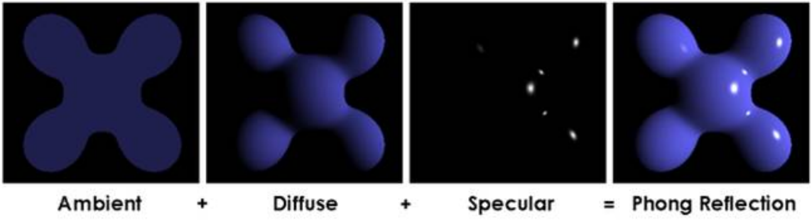
$$\vec{l} + \vec{r} = 2(\vec{l} \cdot \vec{n})\vec{n}$$

$$\vec{r} = 2(\vec{l} \cdot \vec{n})\vec{n} - \vec{l}$$

$$\cos\theta = \vec{r} \cdot \vec{c} = \left(2(\vec{l} \cdot \vec{n})\vec{n} - \vec{l}\right) \cdot \vec{c}$$



Put It All Together



$$C = I_a \cdot C_o + \frac{1 - I_a}{1 + \alpha d + \beta d^2} \left(k_d \cdot \vec{n} \cdot \vec{l} \cdot C_o + (1 - k_d) \cos^s \theta \cdot C_l \right)$$

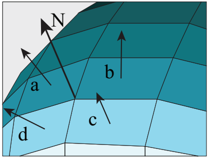
Summary of Parameters

Properties of light	
I_a	Ambient light intensity (0~1)
α, β	Degree of point lighting
C_l	Color of light
$\vec{l}$	Normal vector of light direction

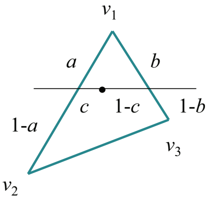
Properties of object	
k_d	Diffusion coefficient ($k_d = 1 - k_s$: specular coefficient)
s	Shininess: how mirror-like
C_o	Color of the object
$\vec{n}$	Normal vector of the plane

Flat, Gouraud, and Phong Shadings

- Flat: one triangle, one color.
- Gouraud: interpolation of vertices colors.
- Phong: interpolation of vertices normal vectors.



Vertex normal



$$(1-c)((1-a)v_1 + av_2) + c((1-b)v_1 + bv_3)$$

Flat, Gouraud, and Phong Shadings

Flat Gouraud Phong