CS 418 Homework 3

out: Tuesday, February 18, 2003
due: Friday, February 28, 2003

Problem 1: Graphics pipeline performance

Referring to the slides from Lecture 10, let's assume that the performance of the graphics pipeline is simply limited by the geometry processing and fragment processing stages. That is, if we keep the geometry processing stage supplied with vertices it will process them at r_v vertices per second, and if we keep the fragment processing stage supplied with fragments it wil get them into the framebuffer at r_f fragments per second.

NVIDIA's marketing literature for the GeForce4 Ti4600, the graphics chip used in the cards in our lab, claims $r_v = 136$ million vertices/sec and $r_f = 1.2$ billion fragments/sec. Ignoring the many factors that can prevent us from actually achieving these numbers, let's take them as given.

We are displaying a scene by compositing 12 layers, each of which is rendered using 70 thousand triangles that tile to cover the entire screen.

1. If each triangle is rendered separately, at how many frames per second can we render this scene for the following framebuffer sizes: 800x600, 1024x768, 1280x1024, 1600x1200?
2. If the triangles are rendered as strips of eight triangles, how may frames per second can be achieved at each resolution?

Problem 2: Super-Hermite splines

Hermite splines are cubic splines defined by point and tangent (that is, value and derivative) constraints at the two endpoints of each segments. More specifically, the following constraints define a Hermite segment:

$$\begin{aligned}\mathbf{p}(0) &= \mathbf{p}_0\\ \mathbf{p}'(0) &= \mathbf{v}_0\\ \mathbf{p}(1) &= \mathbf{p}_1\\ \mathbf{p}'(1) &= \mathbf{v}_1\end{aligned}$$

The resulting spline is defined as follows:

$$\mathbf{p}(t) = \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix}^T \begin{bmatrix} 2 & 1 & -2 & 1 \\ -3 & -2 & 3 & -1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{v}_0 \\ \mathbf{p}_1 \\ \mathbf{v}_1 \end{bmatrix}$$

For added control we decide to add the ability to constrain the center of the curve as well, by introducing another constraint:

$$\mathbf{p}(0.5) = \mathbf{p}_{0.5}$$

1. What degree polynomial is needed to satisfy this additional constraint?
2. Give the matrix equation that defines the new spline (please put the controls in the order $\mathbf{p}_0, \mathbf{v}_0, \mathbf{p}_{0.5}, \mathbf{p}_1, \mathbf{v}_1$). Show your derivation.
3. What happens to the second half of the curve when we move the first endpoint?

Hint: If you formulate your solution as a matrix inversion problem then a computer can do much of the work for you.

Now suppose we want to control the tangent as well as the position at the center point by introducing another constraint:

$$\mathbf{p}'(0.5) = \mathbf{v}_{0.5}$$

4. Give the matrix equation that defines the new spline (please put the controls in the order $\mathbf{p}_0, \mathbf{v}_0, \mathbf{p}_{0.5}, \mathbf{v}_{0.5}, \mathbf{p}_1, \mathbf{v}_1$). Show your derivation.
5. Now what happens to the second half of the curve when we move the first endpoint?
6. How does this new spline differ from a two-segment Hermite spline in terms of (a) locality of control and (b) degree of continuity at the center point?

Problem 3: Cubic basis splines

Uniform cubic B-splines are defined as a sum of translated copies of a single basis function, weighted by the control points. That is, the spline curve is defined by:

$$\mathbf{p}(t) = \sum_i b(t-i)\mathbf{p}_i$$

In class we built up the function b from the box function using recurrence relations and convolution. However, one can also find b more directly by specifying its properties and solving for the function.

We know that the basis function b needs to have the following properties:

- $b(t)$ is zero when $t \leq 0$ or $4 \leq t$.
- b is a piecewise cubic function with its discontinuities at the integers.
- b has C^2 continuity.
- The four overlapping basis functions that define a segment of the spline always sum to 1.

Together, the first two constraints may be summarized as:

$$b(t) = \begin{cases} p_0(t) & \text{if } 0 \leq t \leq 1 \\ p_1(t-1) & \text{if } 1 \leq t \leq 2 \\ p_2(t-2) & \text{if } 2 \leq t \leq 3 \\ p_3(t-3) & \text{if } 3 \leq t \leq 4 \\ 0 & \text{otherwise} \end{cases},$$

where p_k is a cubic polynomial for all k. With this definition, the last constraint is that $p_0 + p_1 + p_2 + p_3 = 1$ (i. e. the sum of the four polynomials is the polynomial 1).

1. Starting from these constraints, prove that the B-spline has the form given in Equations 10-59 and 10-60 in Hearn & Baker.

   Hint: I found it easiest to get p_0 and p_1 first, then p_2 and p_3 by symmetry.

2. In what sense does this show that the cubic B-spline is unique?

3. An order d B-spline has d segments of degree $d-1$ and has C^{d-2} continuity. Does the uniqueness you proved for the cubic ($d = 4$) case hold for all d? Prove your answer.

   Hint: You can answer this in just a few lines.

Problem 4: 3D transformations

If two transformations A and B have the property $AB = BA$ then we say that the two transformations *commute*. In which of the following situations do the 3D affine transformations A and B always commute?

1. A and B are both translations.
2. A and B are both rotations.
3. A and B are both nonuniform scales.
4. A and B are both rotations about the same axis.
5. A is a rotation and B is a uniform scale.
6. A is a rotation and B is a nonuniform scale.
7. A is a rotation about the x axis and B is a nonuniform scale with scale factors a, b, and b along the x, y, and z axes respectively.
8. A is a rotation and B is a translation along the rotation axis.
9. A is a rotation and B is a planar reflection.
10. A is a rotation and B is a reflection across the origin.
11. A is a z-axis shear (see Section 11-4 in Hearn & Baker) and B is a translation along the x axis.
12. A is a z-axis shear and B is a translation along the z axis.

For this problem rotations, scales, shears, and reflections are with respect to the origin and nonuniform scales are aligned with the coordinate axes. A reflection across the origin takes the point $\mathbf{p}$ to the other point that is collinear with $\mathbf{p}$ and the origin and is the same distance away.