Written Assignment #2: Appearances, Curves and Surfaces

15-462 Graphics I, Fall 2003
Doug James
Due: Thursday, October 9, 2003 (**before lecture**)
70 POINTS

October 9, 2003

- The work must be all your own.
- The assignment is due before lecture on Thursday, October 9.
- Be explicit, define your symbols, and explain your steps. This will make it a lot easier for us to assign partial credit.
- Use geometric intuition *together* with trigonometry and linear algebra.
- Verify whether your answer is meaningful with a simple example.

1 Angel, Chapter 6, Exercise 6.14

Show that the halfway vector $\mathbf{h}$ is at the angle at which a surface must be oriented so that the maximum amount of reflected light reaches the viewer.

If we orient the surface so it points in the direction of the halfway vector, then we have equal angles between the halfway vector and the viewer, and between the halfway vector and the light source. This configuration is exactly that of a mirror which reflects all the light towards the viewer.

2 Angel, Chapter 6, Exercise 6.22

Generalize the shadow-generation algorithm (Section 5.10) to handle flat surfaces at arbitrary orientations. (NOTE: This is similar to your second programming assignment.)

Probably the easiest approach to this problem is to transform the given plane to the plane addressed for the simplified shadow algorithm, and also transform the light source and the objects in the same way. After finding the projected shadow points, we can transform everything back.

Notation: Let the quantities of interest be the light position $\mathbf{p_l} = (\mathbf{x_l}, \mathbf{y_l}, \mathbf{z_l}, \mathbf{1})$, the plane $h = (\mathbf{n}, \mathbf{d}) = (\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})$ s.t. $h^T p = 0$, the model point to be projected $\mathbf{p}$, and the projected point $\mathbf{p_p}$.

First, let's translate the problem so that the plane passes through the origin as in our simple algorithm. If we translate in a direction normal to the plane, we must translate by

$$\frac{-d\mathbf{n}}{\mathbf{n^T n}}.$$

Denote this by the 4-by-4 translation matrix, $\mathbf{T}$.

Next, let's rotate the plane so that the plane normal is $\hat{\mathbf{y}}$. This is achieved by the axis-angle rotation with axis $\mathbf{n} \times \hat{\mathbf{y}}$, and angle θ s.t. $\cos\theta = \hat{\mathbf{n}}^\mathbf{T}\hat{\mathbf{y}}$. Denote this by the 4-by-4 rotation matrix, $\mathbf{R}$.

Therefore we have transformed the original problem to the simpler one with parameters:

$$\begin{aligned}
\mathbf{p'_l} &= \mathbf{RTp_l} \quad \text{(Fictitious light point)} \\
\mathbf{p'} &= \mathbf{RTp} \quad \text{(Fictitious model point)} \\
\mathbf{p'_p} &= \mathbf{RTp_p} \quad \text{(Fictitious projected shadow point)}
\end{aligned}$$

Using the new light position $\mathbf{p'_l}$ in our old shadow projection transformation, $\mathcal{P}' = \mathcal{P}(\mathbf{p'_l})$ (see p.263 Angel), we have that

$$\mathbf{p'_p} = \mathcal{P}'\mathbf{p'}$$

or

$$(\mathbf{RTp_p}) = \mathcal{P}'(\mathbf{RTp})$$

so that the actual projected shadow point is

$$\mathbf{p_p} = \mathbf{T^{-1}R^T}\mathcal{P}'\mathbf{RTp}.$$

Therefore the new shadow projection matrix is

$$\mathbf{T^{-1}R^T}\mathcal{P}'\mathbf{RT}.$$

3 Angel, Chapter 7, Exercise 7.3

How is an image produced with an environment map different from a ray-traced image of the same scene?

The major problem is that the environment map is computed without the object in the scene. Thus, all global lighting calculations of which it should be a part are incorrect. These errors can be most noticeable if there are other reflective objects, since these will not show the reflection of the removed object. Other errors can be caused by the removed object no longer blocking light and by its shadows being missing. Other visual errors can be due to distortions in the mapping of the environment to a simple shape, such as a cube, and to errors in a two stage mapping.

4 Angel, Chapter 10, Exercise 10.4

Show that, as long as the four control points for the cubic interpolating curve are defined at unique values of the parameter u, the interpolating geometry matrix always exists.

The matrix for interpolating points $\mathbf{p_0}$, $\mathbf{p_1}$, $\mathbf{p_2}$ and $\mathbf{p_3}$ is

$$\mathbf{A} = \begin{bmatrix} 1 & u_0 & u_0^2 & u_0^3 \\ 1 & u_1 & u_1^2 & u_1^3 \\ 1 & u_2 & u_2^2 & u_2^3 \\ 1 & u_3 & u_3^2 & u_3^3 \end{bmatrix}$$

where

$$\mathbf{p} = \mathbf{Ac}$$

and $\mathbf{c}$ are the polynomial coefficients. The interpolating geometry matrix is $\mathbf{M_I} = \mathbf{A^{-1}}$, and therefore $\mathbf{A}$ must be invertible. The determinant of this $\mathbf{A}$ matrix has the form

$$\det \mathbf{A} = const(u_0 - u_1)(u_0 - u_2)(u_0 - u_3)(u_1 - u_2)(u_1 - u_3)(u_2 - u_3)$$

because if any pair of the interpolating points are the same the matrix will have two identical rows and a zero determinant. By this same reasoning, if the interpolating points are all distinct, then the determinant cannot be zero, and therefore the $\mathbf{A}$ matrix must have an inverse.

5 Angel, Chapter 10, Exercise 10.6

Verify the C^2 continuity of the cubic spline.

10.6 For the cubic B–spline, the blending functions are

$$\mathbf{b}(u) = \frac{1}{6}\begin{bmatrix} (1-u)^3 \\ 4 - 6u^2 + 3u^3 \\ 1 + 3u + 3u^2 - 3u^3 \\ u^3 \end{bmatrix},$$

and the first two derivatives are

$$\mathbf{b}'(u) = \frac{1}{2}\begin{bmatrix} -(1-u)^2 \\ -4u + 3u^2 \\ 1 + 2u - 3u^2 \\ u^2 \end{bmatrix},$$

$$\mathbf{b}''(u) = \begin{bmatrix} 1-u \\ -4 + 6u \\ 1 + 3u + 3u^2 - 3u^3 \\ u^3 \end{bmatrix}.$$

Consider the B–spline $p(u)$ determined by the control points P_{i-2}, P_{i-1}, P_i, and P_{i+1} and next B–spline determined by P_{i-1}, P_i, P_{i+1}, and P_{i+2}. Thus,

$$p(u) = \begin{bmatrix} P_{i-2} & P_{i-1} & P_i & P_{i+1} \end{bmatrix} \mathbf{b}(u)$$

$$q(u) = \begin{bmatrix} P_{i-1} & P_i & P_{i+1} & P_{i+2} \end{bmatrix} \mathbf{b}(u)$$

The values of $p(u)$ and its first two derivatives at $u = 1$ have to be compared with the values of $q(u)$ and its first two derivatives ar $u = 0$. Using the above equations, we find

$$p(1) = q(0) = \frac{1}{6}(P_{i-1} + 4P_i + P_{i+1}),$$

$$p'(1) = q'(0) = \frac{1}{2}(P_{i+1} - P_{i-1}),$$

$$p''(1) = q''(0) = P_{i-1} + 2P_1 + 2P_{i+1}.$$

These results are sufficient to verify C^2 continuity because all the derivatives of polynomials are continuous except possibly at the join points.

6 Angel, Chapter 10, Exercise 10.16

Find the zeros of the Hermite blending functions. Why do these zeros imply that the Hermite curve is smooth in the interval (0,1)?

(Given that the blending functions are just cubic polynomials, and therefore C^∞, the term “smoothness” really refers to how big the “wiggles” in the interpolated curve can be. However, if the blending functions are guaranteed to be nonnegative, then their cubic interpolating linear superposition can undergo only very limited “wiggling.”)

10.16 The four blending functions for the Hermite curve are $(u-1)^2$, $(2u+1)$, $u^2(-2u+3$, $u(1-u)^2$, and $u^2(u-1)$. The zeros of these functions are $(1,1,-1/2)$, $(0,0,-3/2)$, $(0,1,1)$, and $(0,1,1)$. All are either at the edges of the interval (0,1) or outside of it. Each has only a single stationary point in the interval (0,1). Consequently, these blending functions must be smooth inside this interval

7 Angel, Chapter 10, Exercise 10.18

For a 1024×1024 display screen, what is the maximum number of subdivisions that are needed to render a cubic polynomial surface?

Because $1024 = 2^{10} < 1280 < 2048 = 2^{11}$, after at most 11 subdivisions, we are at the resolution of less than a pixel.