Programming #4-3: Part III (4190.562)

Due: June 6, 2018

Given a bicubic Bézier surface

$$S(u,v) = \sum_{i=0}^{3}\sum_{j=0}^{3} \mathbf{b}_{ij} B_i^3(u) B_j^3(v) = \sum_{i=0}^{3}\sum_{j=0}^{3} (x_{ij}, y_{ij}, z_{ij}) B_i^3(u) B_j^3(v),$$

for $0 \le u, v \le 1$, let $N(u,v)$ denote the unit normal vector field of $S(u,v)$:

$$N(u,v) = \frac{S_u(u,v) \times S_v(u,v)}{\|S_u(u,v) \times S_v(u,v)\|},$$

where $S_u(u,v)$ and $S_v(u,v)$ are the u- and v-partial derivatives of $S(u,v)$.

Part I: From an RGB image of size $(m+1)\times(n+1)$, convert the image into a discrete height field $h(i,j)$, $(i = 0, \cdots, m$ and $j = 0, \cdots, n)$, and generate a quad mesh of mn quadrangles and $(m+1) \times (n+1)$ vertices:

$$Q(i,j) = S\left(\frac{i}{m}, \frac{j}{n}\right) + h(i,j) \cdot N\left(\frac{i}{m}, \frac{j}{n}\right).$$

When we edit the bicubic Bézier surface $S(u,v)$, each point of the quad mesh $Q(i,j)$ should move to the corresponding new position under the shape deformation of $S(u,v)$.

Part II: For a constant radius $r > 0$ and $m, n = 1024$, generate a quad mesh of mn quadrangles and $(m+1) \times (n+1)$ vertices:

$$Q(i,j) = S\left(\frac{i}{m}, \frac{j}{n}\right) + r \cdot N\left(\frac{i}{m}, \frac{j}{n}\right),$$

which is an approximation to the offset surface of $S(u,v)$. Now consider a uniform hierarchy of the quadrangles. For $h = 1, \cdots, 10$, and $k_1, k_2 = 0, \cdots, 2^{10-h} - 1$, bound each submesh of $2^h \times 2^h$ quadrangles with vertices $Q(k_1 2^h + i, k_2 2^h + j)$, $(i, j = 0, \cdots, 2^h)$, using a TSS with thickness $\epsilon > 0$ from the tetrahedron with four corners:

$$Q(k_1 2^h, k_2 2^h), Q(k_1 2^h + 2^h, k_2 2^h), Q(k_1 2^h + 2^h, k_2 2^h + 2^h), Q(k_1 2^h, k_2 2^h + 2^h).$$

Part III: Given two bicubic Bézier surfaces $S_1(u,v)$ and $S_2(s,t)$, $(0 \le u, v, s, t \le 1)$, and $n = 1024$, generate quadmeshes Q_1 and Q_2, each with n^2 quadrangles and $(n+1)^2$ vertices:

$$Q_1(i,j) = S_1\left(\frac{i}{n}, \frac{j}{n}\right), \quad Q_2(k,l) = S_2\left(\frac{k}{n}, \frac{l}{n}\right), \quad (\text{for } i, j, k, l = 0, \cdots, n),$$

which are approximations to the surfaces of $S_1(u,v)$ and $S_2(s,t)$. Using the GJK algorithm applied to the TSS trees for Q_1 and Q_2, compute the intersection curve $Q_1 \cap Q_2$.