CS480/CS680 Problem Set 2 Model Solution

(100 pts / 120 pts)

1. (15 points)

Algorithm

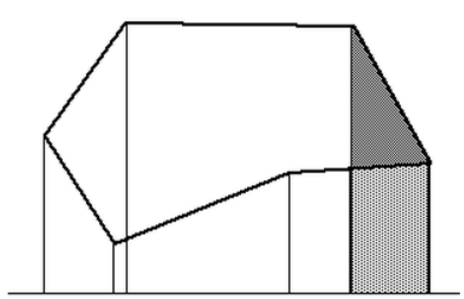
```
// Algorithm to determine whether polygon vertices are given
// in CW or CCW order for a front facing 2D simple polygon

Input: v1, …, vN                    // N polygon vertices
Output: True or False               // True if CCW, False if CW

float sum = 0;

for (j=1; j<=N; ++j){
   if(j==N)
         k=1;
   else
         k=j+1;

   // adding area under vj -> vk to our accumulating sum
   sum += (vj.x vk.y – vk.x vj.y);
}

// check the sign of sum (double the area)
return (sum > 0.0);
```

Explanation

[By Paul Bourke: http://paulbourke.net/geometry/polygonmesh/]

The sign of the area expression above (without the absolute value) can be used to determine the ordering of the vertices of the polygon. If the sign is positive then the polygon vertices are ordered counter clockwise about the normal, otherwise clockwise.

To derive this solution, project lines from each vertex to some horizontal line below the lowest part of the polygon. The enclosed region from each line segment is made up of a triangle and rectangle. Sum these areas together noting that the areas outside the polygon eventually cancel as the polygon loops around to the beginning.



2. (15 points)

<u>Solution 1</u>

$$\mathbf{I_m} = \sqrt{\frac{m^2 + 1}{2}}$$

<u>Solution 2</u>

The intensity at m=1 is: $\mathbf{I_1 = 1.0}$

The intensity at m=0 is: $\mathbf{I_0} = \frac{\mathbf{1.0}}{\sqrt{\mathbf{2}}}$

We can now interpolate the color intensity of intermediate slopes as follows:

$$\Theta = \tan^{-1} m$$

$$\alpha = \frac{\Theta}{45}$$

$$I_m = (1-\alpha)\, I_0 + \alpha\, I_1$$

3.

(a) (15 points)

Giving a counter-example will suffice. Assume:

Angle of rotation $\theta = \pi/2$
Two rotation axes $\mathbf{u}_1 = (1,0,0)$ and $\mathbf{u}_2 = (0,1,0)$
Original point $\mathbf{P} = (0,1,1,1)$
$\mathbf{P}'$ is the point after performing rotation about $\mathbf{u}_1$ followed by rotation about $\mathbf{u}_2$
$\mathbf{P}''$ is the point after performing rotation about $\mathbf{u}_2$ followed by rotation about $\mathbf{u}_1$

The quaternion representing the rotation θ around $\mathbf{u}_1$ is:

$$\mathbf{q}_1 = \left(\cos\frac{\theta}{2},\ \mathbf{u}_1 \sin\frac{\theta}{2} \right) = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0 \right)$$

The quaternion representing the rotation θ around $\mathbf{u}_2$ is:

$$\mathbf{q}_2 = \left(\cos\frac{\theta}{2},\ \mathbf{u}_2 \sin\frac{\theta}{2} \right) = \left(\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}, 0 \right)$$

If we apply $\mathbf{q}_1$ rotation followed by $\mathbf{q}_2$ rotation to the point P, we get the following P':

$$\mathbf{q}_2\ \mathbf{q}_1 = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, -\frac{1}{2} \right)$$

$$\mathbf{q}_1^{-1}\ \mathbf{q}_2^{-1} = \left(\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}, \frac{1}{2} \right)$$

$$\mathbf{P}' = \mathbf{q}_2\ \mathbf{q}_1\ \mathbf{P}\ \mathbf{q}_1^{-1}\ \mathbf{q}_2^{-1} = (\ 0,\ 1, -1, -1\ )$$

If we apply $\mathbf{q}_2$ rotation followed by $\mathbf{q}_1$ rotation to the Point P, we get the following P'':

$$\mathbf{q}_1\ \mathbf{q}_2 = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \right)$$

$$\mathbf{q}_2^{-1}\ \mathbf{q}_1^{-1} = \left(\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2} \right)$$

$$\mathbf{P}'' = \mathbf{q}_1\ \mathbf{q}_2\ \mathbf{P}\ \mathbf{q}_2^{-1}\ \mathbf{q}_1^{-1} = (\ 0,\ 1,\ 1,\ 1\ )$$

$\mathbf{P}' \neq \mathbf{P}''$. Therefore, in general, two 3D rotations about different rotation axes do not commute.

(b) (15 points)

Two rotations are commutative under the condition that they are rotations about the same axis. We derive this conclusion as follows:

Using the quaternion representation, applying $\mathbf{q}_1$ followed by $\mathbf{q}_2$ to a point $\mathbf{P}$ yields:

$$\mathbf{P}' = \mathbf{q}_2\ \mathbf{q}_1\ \mathbf{P}\ \mathbf{q}_1^{-1}\ \mathbf{q}_2^{-1}$$

$$= (\mathbf{s}_2,\ \mathbf{v}_2)(\mathbf{s}_1,\ \mathbf{v}_1)\mathbf{P}(\mathbf{s}_1,\ -\mathbf{v}_1)\ (\mathbf{s}_2,\ -\mathbf{v}_2)$$

$$= \mathbf{q}_2\ \mathbf{q}_1\ \mathbf{P}(\mathbf{q}_2\ \mathbf{q}_1)^{-1}$$

Alternatively, switching the order of the rotations yields:

$$\mathbf{P}' = \mathbf{q}_1\ \mathbf{q}_2\ \mathbf{P}\ \mathbf{q}_2^{-1}\ \mathbf{q}_1^{-1}$$

$$= (\mathbf{s}_1,\ \mathbf{v}_1)(\mathbf{s}_2,\ \mathbf{v}_2)\mathbf{P}(\mathbf{s}_2,\ -\mathbf{v}_2)\ (\mathbf{s}_1,\ -\mathbf{v}_1)$$

$$= \mathbf{q}_1\ \mathbf{q}_2\ \mathbf{P}(\mathbf{q}_1\ \mathbf{q}_2)^{-1}$$

The above two rotations are only equivalent when:

$$\mathbf{q}_1\ \mathbf{q}_2 = \mathbf{q}_2\ \mathbf{q}_1$$
$$(\mathbf{s}_1\ \mathbf{s}_2 - \mathbf{v}_1\ \mathbf{v}_2\ ,\ \mathbf{s}_1\ \mathbf{v}_2 + \mathbf{s}_2\ \mathbf{v}_1 + \mathbf{v}_1 \times \mathbf{v}_2) = (\mathbf{s}_2\ \mathbf{s}_1 - \mathbf{v}_2\ \mathbf{v}_1\ ,\ \mathbf{s}_2\ \mathbf{v}_1 + \mathbf{s}_1\ \mathbf{v}_2 + \mathbf{v}_2 \times \mathbf{v}_1)$$

For this equality to hold, $\mathbf{v}_2 \times \mathbf{v}_1 = \mathbf{v}_1 \times \mathbf{v}_2$ must be satisfied. There are two possible scenarios for the commutability of the cross-product, one is either one of the two vectors is zero and the other is $\mathbf{v}_1 = \mathbf{v}_2$. Geometrically, the first scenario involves rotations that are multiples of 2π about the axis, which brings the object back to its original position. In this case, applying rotations $\mathbf{q}_1$ and $\mathbf{q}_2$ is equivalent to just applying the one with non-zero $\mathbf{v}$ component. The second scenario corresponds to two rotations about the same axis.

4. (15 points)

Step 1: Translate such that the line's y-intercept (c) becomes the canonical origin

$$\mathbf{T_{in}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -c \\ 0 & 0 & 1 \end{bmatrix}$$

Step 2: Rotate such that the line coincides with the canonical x-axis

$$\mathbf{R_{in}} = \begin{bmatrix} cos\ \boldsymbol{\theta} & sin\ \boldsymbol{\theta} & 0 \\ -sin\ \boldsymbol{\theta} & cos\ \boldsymbol{\theta} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Step 3: Reflect about the x-axis

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Step 4: Rotate such that the line is back to its original slope

$$\mathbf{R_{out}} = \begin{bmatrix} cos\ \boldsymbol{\theta} & -sin\ \boldsymbol{\theta} & 0 \\ sin\ \boldsymbol{\theta} & cos\ \boldsymbol{\theta} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Step 5: Translate back such that the original y-intercept (c) is restored

$$\mathbf{T_{out}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & c \\ 0 & 0 & 1 \end{bmatrix}$$

5. (10 points)

x−direction shear, example amount: sh_x= 0.5
reference line, y_{ref}: 4

matrix formulation: $\begin{bmatrix} 1 & sh_x & -sh_x * y_{ref} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

matrix values: $\begin{bmatrix} 1 & 0.5 & -2 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

6. (15 points)

Step 1: Build a **uvn** coordinate system

$$\mathbf{v} = \frac{v}{|v|}$$

$$\mathbf{u} = \frac{(v_y, -v_x, v_z)}{|v|}$$

$$\mathbf{n} = \mathbf{u} \times \mathbf{v}$$

Step 2: Translate such that **C** coincides with the canonical origin

$$\mathbf{T_{in}} = \begin{bmatrix} 1 & 0 & 0 & -C_x \\ 0 & 1 & 0 & -C_y \\ 0 & 0 & 1 & -C_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Step 3: Transform **uvn** basis to **xyz** basis

$$\mathbf{R_{in}} = \begin{bmatrix} \frac{u_x}{|u|} & \frac{u_y}{|u|} & \frac{u_z}{|u|} & 0 \\ \frac{v_x}{|v|} & \frac{v_y}{|v|} & \frac{v_z}{|v|} & 0 \\ \frac{n_x}{|n|} & \frac{n_y}{|n|} & \frac{n_z}{|n|} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Note: If u, v, and n are unit vectors as derived in step 1, then we do not need to divide the elements of this matrix by the magnitude of the relevant vectors.

Step 4: Scale along the **v**-axis (now coinciding with the y-axis)

$$\mathbf{S} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & S_v & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Step 5: Rotate **xyz** basis to **uvn** basis

$$\mathbf{R}_{\text{out}} = \begin{bmatrix} \frac{u_x}{|u|} & \frac{v_x}{|v|} & \frac{n_x}{|n|} & 0 \\ \frac{u_y}{|u|} & \frac{v_y}{|v|} & \frac{n_y}{|n|} & 0 \\ \frac{u_z}{|u|} & \frac{v_z}{|v|} & \frac{n_z}{|n|} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Note: If u, v, and n are unit vectors as derived in step 1, then we do not need to divide the elements of this matrix by the magnitude of the relevant vectors.

Step 6: Translate back such that C is back to its original location

$$\mathbf{T}_{\text{out}} = \begin{bmatrix} 1 & 0 & 0 & C_x \\ 0 & 1 & 0 & C_y \\ 0 & 0 & 1 & C_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The required 3D homogeneous transformation matrix is: $M = \mathbf{T}_{\text{out}}\ \mathbf{R}_{\text{out}}\ \mathbf{S}\ \mathbf{R}_{\text{in}}\ \mathbf{T}_{\text{in}}$

7. (CS 680 only; 20 points)

- Select an origin for the plane coordinate system

  Select any point (x_0, y_0, z_0) s.t. $ax_0 + by_0 + cz_0 = d$ (i.e. any point on the plane). This point will be treated as the origin of the plane coordinate system. To map this point to the xyz canonical origin we need the following translation:

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & -x_0 \\ 0 & 1 & 0 & -y_0 \\ 0 & 0 & 1 & -z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Find the mapping between the canonical xyz and the uvn

  Getting an orthogonal basis for the plane:

  u = (a, b, c) the normal to the plane

  v = (-b, a, 0) a vector orthogonal to the plane normal

  n = v x u a vector orthogonal u and v

  Mapping the canonical xyz to uvn:

$$\mathbf{R}_{out} = \begin{bmatrix} \frac{\boldsymbol{n}}{|\boldsymbol{n}|} \Big| \frac{\boldsymbol{v}}{|\boldsymbol{v}|} \Big| \frac{\boldsymbol{u}}{|\boldsymbol{u}|} & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{R}_{in} = \mathbf{R}_{out}^{-1} = \mathbf{R}_{out}^{T}$$

- Find the reflection transformation with respect to the canonical xy-plane

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Finally, the required transformation matrix is:

$$\mathbf{M} = \mathbf{T}^{-1}\, \mathbf{R}_{out}\, \mathbf{Q}\, \mathbf{R}_{in}\, \mathbf{T}$$