CS452 Exam 2

Lewis Collum

Updated: April 23, 2020

1

(a)

Phong reflection is the summation of the ambient, diffuse and specular reflection terms. It is used to represent how light reflects off an object surface.

(d)

A. For point lights and spotlights describe the intensity at a point using a factor of $1/(a+bd+cd^2)$ instead of $1/d^2$, where a, b, and c are user-specified constants.

B. For specular lighting, use the halfway vector between i and v and then compute $(n \cdot h)^{\alpha'}$ instead of $(r \cdot v)^{\alpha'}$.

(c)

Cube and sphere mapping.

2

(a)

(1) and (2). (2) is a vanilla example of a B-spline. (1) can be achieved by replicating points. Replicating points pulls the curve toward the replicated points. In (1), the points that are highly replicated (relative to the other points) are the ones that appear sharp.

Furthermore, for (3), the curve lies outside the convex hull and cannot be achieve with a B-spline.

(b)

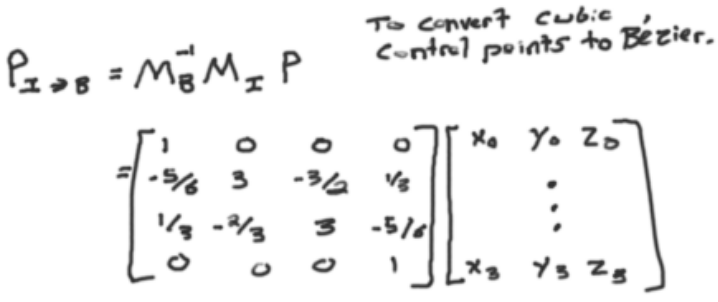
Neither. Diffuse reflection scatters light evenly in all direction assuming $\cos(\theta) > 0$.

3

(a)

Use Recursive Subdivision. That is recursively find the midpoint of each segment connected (p_0p_1, p_1p_2, etc.) and then create new segments that connect each midpoint. What remains is a curve that resembles a bezier curve.

(b)



4

(bonus)

```python
import numpy as np

e = np.array([0, 10, 2])
a = np.array([0, 0, 2])
up = np.array([0, 0, 1])

def normalize(vector):
    return vector/np.sqrt(np.sum(vector**2))

d = e - a
n = normalize(d)
u = normalize(np.cross(up, n))
v = normalize(np.cross(n, u))

rotation = np.array([
    [u[0], v[0], n[0], 0],
    [u[1], v[1], n[1], 0],
    [u[2], v[2], n[2], 0],
    [0, 0, 0, 1]])

translation = np.array([
    [1, 0, 0, e[0]],
    [0, 1, 0, e[1]],
    [0, 0, 1, e[2]],
    [0, 0, 0, 1]])

modelView = np.dot(rotation, translation)

print(f'n = {n}')
print(f'u = {u}')
print(f'v = {v}')
print(f'modelView = \n{modelView}')
```

```
n = [0. 1. 0.]
u = [-1.  0.  0.]
v = [ 0. -0.  1.]
modelView =
[[-1.  0.  0.  0.]
 [ 0.  0.  1.  2.]
 [ 0.  1.  0. 10.]
 [ 0.  0.  0.  1.]]
```

5

(a)

Answer: (iii)

The triangle shown on the left is a projection of the rotated triangle shown in (iii).

(b)

A. Determine d_1 by making a shadow map. A shadow map is created from the viewpoint of light and instead of containing colors, it contains the depth to each pixel.
B. Compute d_2 during shading and compare d_2 to d_1. In general, if $d_2 > d_1$ then the corresponding pixel is rendered as being in a shadow. Due to floating point comparison issues, a more realistic comparison might be $d_2 > d_1 +$ `<some small float>`.