Name: SOLUTIONS

Midterm Exam

15-462 / 15-662 Computer Graphics, Spring 2014

60 points
Closed book, closed notes, no calculators. Use the back of the sheets for extra space.
Write down your assumptions and steps so that I can give you partial credit.

1. **(10 points) BRDFs and Surface Reflectance.** Measuring a BRDF for an anisotropic surface may involve sampling inputs and outputs over 4 total dimensions: two dimensions may parameterize the incoming light source direction, and two dimensions may parameterize the outgoing light source direction. Suppose we are certain that the surface to be measured is an ideal diffuse (i.e., Lambertian) surface.

Perfect diffuse surface

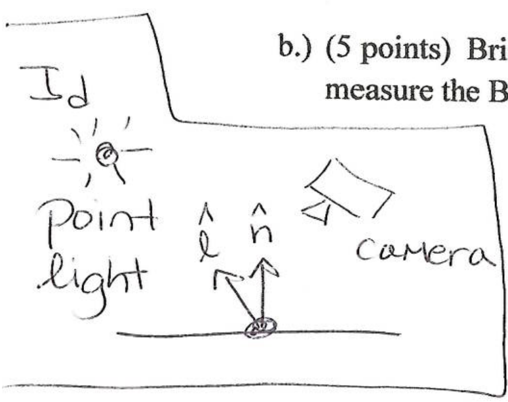
$I = K_d I_d (n \cdot l)$

a.) (5 points) What is the minimum number of dimensions we must sample to fully capture the BRDF of this surface? Explain your answer.

$\boxed{\text{Zero}}$

We can make a $\boxed{\text{single measurement}}$ and solve directly ~~& solve for~~ for $\boxed{K_d = \frac{I}{I_d (n \cdot l)}}$

b.) (5 points) Briefly describe a simple hardware setup that would allow you to measure the BRDF of an ideal diffuse (Lambertian) surface.



Carefully measure $\hat{l}$. Take a single picture
Solve for k_d as above, for red, green, and blue channels separately.

Practically, taking multiple samples & averaging is better.

2. **(5 points) OpenGL Depth Buffer.** True or false: Under perspective projection, the depth information stored in the OpenGL depth buffer is a linear scaling of the true distance of an object from the camera. Clearly and concisely explain your answer.

$\boxed{\text{False}}$

By construction, the perspective Projection matrix preserves $\boxed{\text{depth ordering}}$ but does not preserve $\boxed{\text{relative depth}}$.

Name: SOLUTIONS

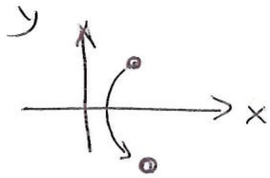
3. **(5 points) Shaders and GLSL.** List one thing that you can do in a fragment shader that cannot be done in a vertex shader. Clearly and concisely explain your answer.

$\boxed{\text{Toon shading}}$ (or other reasonable answer)

As a triangle is rasterized, its color would ordinarily be computed using barycentric coordinates (Gouraud shading). In a fragment shader, we can map the Gouraud shading color into a toon color map having only 2 colors (for example). This operation must be done per fragment, not per vertex, to achieve sharp color boundaries within a triangle.

4. **(10 points) 2D Transforms.**

a.) (5 points) Give a 2x2 matrix that reflects (mirrors) any 2D point about the x-axis.



$$\begin{bmatrix} a \\ b \end{bmatrix} \rightarrow \begin{bmatrix} a \\ -b \end{bmatrix}$$

$$\boxed{\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}}$$

b.) (5 points) Is the 2x2 matrix which mirrors points about the x-axis a rotation matrix? Why or why not?

$\boxed{\text{No.}}$

Its determinant is -1

– OR –

It represents a left-handed coordinate system

$$\begin{array}{c} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{array} \begin{array}{l} x \\ -y \end{array}$$

$X \times -y = -z$

LEFT HANDED

Name: SOLUTIONS

5. **(5 points) Splines**. Why are cubic splines typically preferred to higher order (e.g., 4^{th} or 5^{th} or n^{th} order) polynomials in computer graphics?

5 – compare both
4 – good cubic, not clear on why not use higher
3 – mention wiggliness, but not explain how cubic splines solve this

2 – vague
1 – only mentions computational efficiency

Higher order polynomials can have shapes that are difficult to predict and control (ie., they can be wiggly).

Cubic splines are sufficient to allow interpolation and C^1 continuity & are easier to predict & control.

6. **(5 points) 3D Transformations and Normal Vectors**. Suppose we are given a transformation matrix M that was created by combining a number of rigid body transformations (i.e., some sequence of translations and rotations). True or false: both vertex positions and surface normals can be accurately transformed by multiplying them by matrix M. Concisely explain your answer.

TRUE.

Rigid body transformations preserve distances between points & angles between vectors. Thus, they can be used for normals as well as for vertices.

7. **(5 points) Rendering Reflections.** Suppose we want to render an accurate reflection in a large mirror. True or false: Ray tracing is in general a better choice than OpenGL with an environment map. Concisely explain your answer.

TRUE

Using ray tracing, we can trace all rays from the eye that bounce off the mirror as perfect reflection rays and pick up the perceived color from the next object those rays strike.

An environment map, in contrast, cannot store accurate reflections for every eye/mirror point pair.

Name: SOLUTIONS

8. **(20 points) Implicit functions and distance.** Certain tasks in graphics benefit from an ability to quickly calculate the distance to the closest object. We are going to explore calculating the distance to a plane using implicit functions. For all of the questions below, assume you are given a plane defined by the implicit function

$$\mathbf{f(x,y,z) = 2x + y - 3z} \cancel{= 0} + 1 = 0$$

a.) (5 points) Choose some specific point **r** that lies on the plane. Test your point by ensuring that $\mathbf{f(r_x, r_y, r_z) = 0}$.

$$\boxed{r = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}}$$

(There are many answers — test that $f(r_x, r_y, r_z) = 0$)

b.) (5 points) Give a vector **n** that is normal to the plane. It does not have to be a unit vector.

$$\boxed{n = \begin{bmatrix} 2 \\ 1 \\ -3 \\ 0 \end{bmatrix}}$$

(Any scaled version of this n is fine.)

c.) (5 points) Using **r** and **n** from parts a.) and b.), write an expression for a point **s** that lies at unit distance from the plane. In other words, the distance of point **s** from the plane must be equal to 1.

$$\boxed{s = r + \frac{n}{\|n\|}}$$

d.) (5 points) Use implicit function f(x,y,z) defined above to create a distance function D(x,y,z) that exactly computes the signed distance of a point $[x, y, z, 1]^T$ from the given plane.

$$\boxed{D(x,y,z) = \frac{f(x,y,z)}{f(s_x, s_y, s_z)}}$$