CS 465 Prelim 1

Tuesday 4 October 2005—1.5 hours

Problem 1: Image formats (18 pts)

1. Give a common pixel data format that uses up the following numbers of bits per pixel: 8, 16, 32, 36. For instance, given 24 bits per pixel you might answer "8 bits for each of red, green, and blue."

   Solution

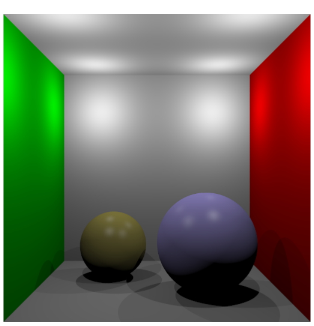
   Any reasonable answers were excepted as long as they were the correct number of bits. Some canonical answers are: 8 bits - single channel gray scale; 16 bits - RGB, 5 red, 6 green, 5 blue; 32 bits - RGBA, 8 red, 8 green, 8 blue, 8 alpha; and 36 bits - 12 red, 12 green, 12 blue.

2. Explain an advantage of a display system that uses nonlinear intensity quantization (that is, one that requires gamma correction) over a display that uses linear quantization, given that the purpose of the display is to show images to a human viewer.

   Solution

   Humans are sensitive to relative changes in intensity. This means that for darker shades we are more sensitive to changes in absolute intensity. Unlike a linear system, a nonlinear quantization has more quanta in the darker regions than lighter regions, but this is desirable because this is where humans can detect finer differences.

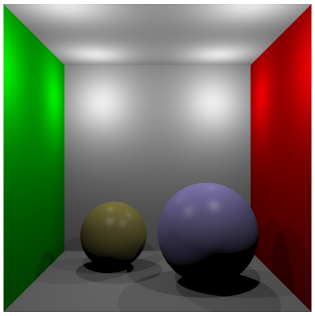
Problem 2: Ray tracing bugs (36 pts)



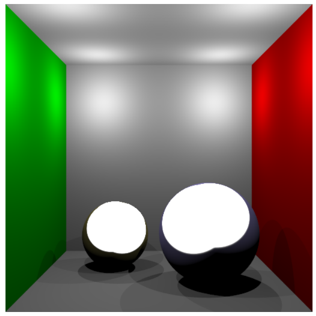
Here is a correct ray-traced rendering of a scene consisting of a five-sided cube containing two spheres lit by four lights near the top corners of the box. The size of the cube is two units on a side.

We modified the ray tracer to introduce the bugs described on the next page. Match the bugs with the corresponding incorrect outputs. For each case, give a brief explanation of how the bug leads to the visible changes in the image.

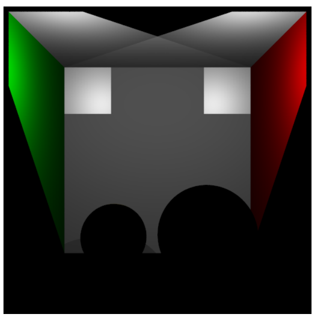
For example, the image labeled “Example” matches the bug “Failing to normalize the half vector in Blinn-Phong shading.” A suitable explanation is “If the half vector is not normalized its length will sometimes be greater than 1. This can result in the dot product with the normal being greater than 1, and when this number is taken to a large power it becomes very large, leading to the very bright areas on the spheres.”



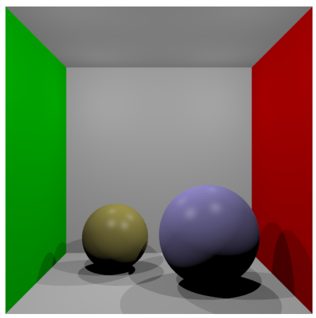
Correct



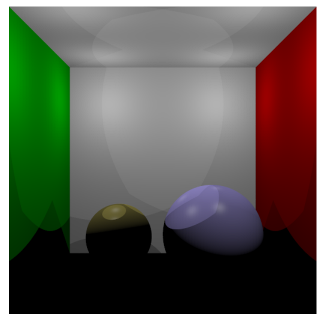
Example



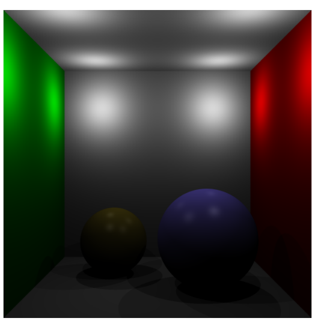
A



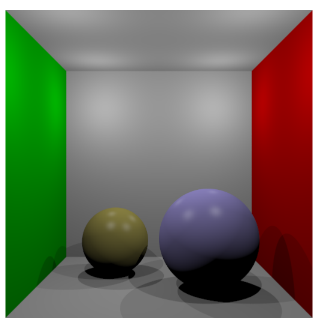
B



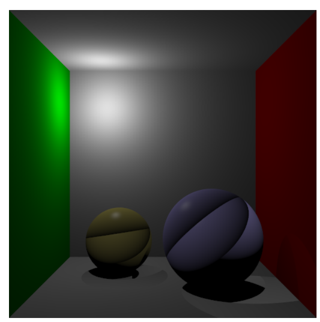
C



D



E



F

0. *Example*: Failing to normalize the half vector in Blinn-Phong shading

1. Omitting gamma correction

   Solution

   D. Gamma correction raises the color to a fractional exponent. Since the colors themselves are all fractional, this operation increases the value of all the colors. Then omitting the correction will make the image darker and it will make it more dark in the darkest regions of the correct image.

2. Overwriting the shading result for each light source, rather than adding

   Solution

F. Each point is colored only by the last visible light resulting in discontinuities in shading along the shadow boundaries for the lights.

3. Omitting the $1/r^2$ falloff for light sources

   Solution

   B. Without the falloff, the light's intensity seems constant at every point. This makes the shading constant for flat surfaces and the hot spots from the lights will disappear.

4. Assuming shadow rays' directions are always normalized when in fact they are not

   Solution

   C. The t value of the ray is used to determine whether a shadow ray intersects something before the light source. If the ray is not normalized, the t values will vary depending on the length of the direction vector. This causes the shadow boundaries to change and making shadows where they should not be.

5. Forgetting to normalize the light vector in shading computations

   Solution

   E. The dot product between two vectors equals the cosine of the angle between them only if the vectors are normalized. Otherwise, the product is proportional to the lengths of the input vectors. When the direction vector is longer than 1 this scaling makes the dot product larger, and if its shorter it makes the product smaller. When this happens smoothly it causes the variation of shading based on distance from the light to be smoothed out, but makes the lower part of the scene brighter since there the light vectors are longer than 1.

6. Failing to end shadow rays at the light source

   Solution

   A. If the shadow rays are not stoped at the light, objects behind the light source can cause it to be occluded. In this scene, the only direction where there is no geometry is towards the camera, so only areas where the shadow ray would pass out of the front of the box are unoccluded.

Don't forget the explanations!

Problem 3: Sampling and reconstruction (22 pts)

1. Suppose you are using a reconstruction filter for translating images around in the plane without changing their size, and it performs well for that application. For each of the following other applications, will it be necessary to make the filter larger, make it smaller, or keep it the same size relative to the input pixel grid?

   (a) Reducing the image (downsampling)

   Solution

   When resampling images, the filter size should be at least as large is the area covered in the input image by an output pixel. When reducing, each output pixel covers more are of the input image, so the filter would need to be made larger.

   (b) Enlarging the image (upsampling)

   Solution

   When enlarging, the output pixels are smaller than the input pixels. However, making the filter smaller is not advisable because it may not cover any of the input pixel samples. In this case, the filter should stay the same size.

2. Suppose I am antialiasing an image in a ray tracer by supersampling it using a particular filter. Despite my using that filter, the image still has stair-steps on the edges of objects.

   (a) In what way do you need to change the filter to reduce this problem?

   Solution

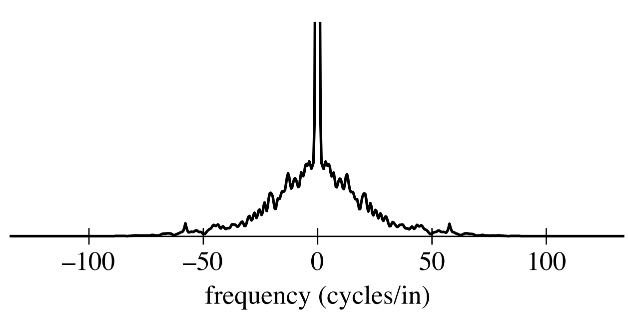
   Use a larger or smoother filter.

   (b) In what way will the image be degraded if you go too far?

   Solution

   If the filter is large enough, it can make the image appear blurred and indistinct.

3. Suppose you are scanning a photographic print. The intensity of the photograph along any horizontal or vertical line has a Fourier transform that looks about like this:



Your scanner can operate at various resolutions, but it always point samples the image. Should you use 50, 100, or 200 samples per inch to achieve the most compact result without introducing aliasing artifacts? How can you tell?

Solution

The Nyquist theorem says that the sampling rate should be twice the largest frequency in the input signal. Since this image contains frequencies above 50 Hz, the 100 Hz sampling rate is not sufficient. Two hundred samples per inch must be used.

Problem 4: Compositing (24 pts)

Suppose we have an image of a gray elephant, with an alpha matte to delineate foreground from background. The image is stored with pre-multiplied alpha.

1. If we accidentally use the image in a program that expects non-premultiplied alpha, will the edges come out too dark, about right, or too light if the background is:

   (a) solid black

   (b) solid white

   (c) about the same color as the elephant

   Explain why by writing down the equations for the correct and incorrect results.

   Solution

   Correct compositing uses the following equation.

   $$C_{\text{output}} = \alpha C_{\text{foreground}} + (1 - \alpha) C_{\text{background}}$$

   When alpha is pre-multiplied, the foreground image is assumed to store $\alpha C_{\text{foreground}}$ and forgetting this means that α is multiplied by the foreground color twice, resulting in the equation:

   $$C_{\text{output}} = \alpha^2 C_{\text{foreground}} + (1 - \alpha) C_{\text{background}}$$

   Since α is strictly less than 1, this incorrect equation produces outputs that are strictly less than the correct equation. Thus in all cases the output will be darker.

2. If we do the compositing correctly but overlay the image twice in the same place, will the edges come out too dark, about right, or too light if the background is:

   (a) solid black

   (b) solid white

   (c) about the same color as the elephant

   Explain why informally. There is no need to write down the equations for this part (we did not find them all that helpful).

   Solution

   The easiest way to think about this is that in the edge regions, the elephant is getting composited twice, so the colors here will be closer to the elephant than the background. On a black background, the elephant is lighter so the edges are lighter, on the white, they are darker, and on the gray, they are about the same.

Assume in all cases we are using the **over** operation.