CS 465 Homework 1

out: Monday 27 August 2007
due: Monday 3 September 2007

Problem 1: Miscellaneous Math (Chapter 2 of Shirley)

1. *Implicit functions:* Give an implicit function $f_c(\mathbf{x}; r_c)$, where $\mathbf{x} = (x, y) \in \mathbb{R}^2$, to represent a circle of radius r_c centered at the origin?
2. *Implicit functions:* Give an implicit function $f_s(\mathbf{x}; r_s)$ to represent a square of side length $2r_s$ centered at the origin.
3. *Parametric curves:* Propose a parametric curve representation $\mathbf{p}_c(\theta)$ for the aforementioned circle. Assume that the parameter is $\theta \in [0, 2\pi]$.
4. *Parametric curves:* Propose a parametric curve representation $\mathbf{p}_s(\theta)$ for the aforementioned square. Assume that the parameter is also $\theta \in [0, 2\pi]$.
5. *Morphing:* Shape morphing allows one to define smoothly varying geometric transitions between two or more shapes. Use linear interpolation to define morphs between a circle (of radius r_c) and a square (of "radius r_s) using implicit function representations as input and output. Assume that the morphing parameter is $\alpha \in [0, 1]$ with $\alpha = 0 \leftrightarrow$ circle, and $\alpha = 1 \leftrightarrow$ square.
6. *Morphing:* Use linear interpolation to define morphs between a circle (of radius r_c) and a square (of "radius" r_s) using parametric curve representations as input and output.
7. *Morphing:* Describe how parametric and implicit shape representations compare for morphing between objects of different topology. For example, how would you morph between a sphere and a torus? Which representation is better for morphing?

Problem 2: Image Representation

Suppose you have an image editor which stores each pixel in three double-precision floating point numbers, using the convention that pixel values are linearly related to display intensity, with the value 1.0 representing the maximum displayable intensity. Assume you are editing an image of size 1600x1200 pixels.

[In your answers, follow tradition by giving data sizes in kilobytes, megabytes, or gigabytes, and data rates in kilobits, megabits, or gigabits per second. Don't forget that these units refer to multiples of 1024, not multiples of 1000. Round your answers to three significant digits—for example, 2.47 MB.]

1. How much memory will the image occupy?

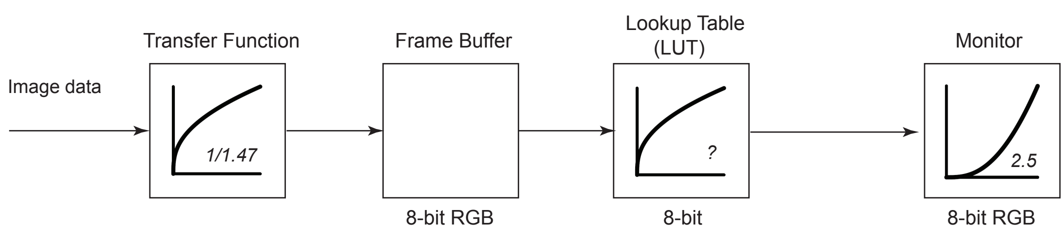
2. If the editor has to export the image as a 8-bit RGB (24-bit color) file, what conversions must the application do? Assume the standard functions *round*, *pow*, *min*, and *max* are available.

3. If the file is not compressed, how much disk space will it occupy?

4. If the monitor of the system has a resolution of 1280x1024. It is working at a depth of 24-bit color and a refresh rate of 75Hz, at what data rate is information flowing across the cable to the monitor?

Problem 3: Gamma Correction

The following questions are based on the following display scheme which are used in several systems, as in the following figure. A full-screen RGB color image is displayed on the screen by transferring it to a framebuffer in the graphics system, which causes it to be stored in the memory on the graphics card. The graphics hardware reads out this image and transmits it via a digital link to an LCD display.



Before the image is stored in the framebuffer, it goes through a transfer function so that it is stored in the framebuffer with a nonlinear quantization at gamma 1.47. The framebuffer uses an 8 bit integer format. The graphics hardware has to convert the format of the framebuffer to the display's pixel format. This is done using a simple lookup table (LUT).

The monitor's display gamma is 2.5 and has a maximum displayable intensity of I_{max} and is in an environment where viewing flare is 1%—that is, the pixel value zero results in an intensity of $0.01 I_{\text{max}}$, and the maximum brightness is $1.01 I_{\text{max}}$.

1. If we want the image pixel values to be linearly related to display intensity, what should be the first 5 entries of the graphics card's lookup table? Round carefully to ensure that each pixel value gets mapped to the displayable value that comes closest to the intensity it's supposed to represent. Assume the graphics hardware does not correct for viewing flare.

2. What is the highest-contrast step in the display's tone scale? Your answer will read "the worst step is from pixel value N to $N + 1$, where the luminance increases by $X\%$." How much viewing flare does it take for the worst step to be invisible, assuming a just noticeable difference of 2%?

3. Consider that the graphics card has a 16-bit/color integer framebuffer instead of 8-bit/color, but the monitor can still only display 8-bit RGB color. Answer the former question again in this case.