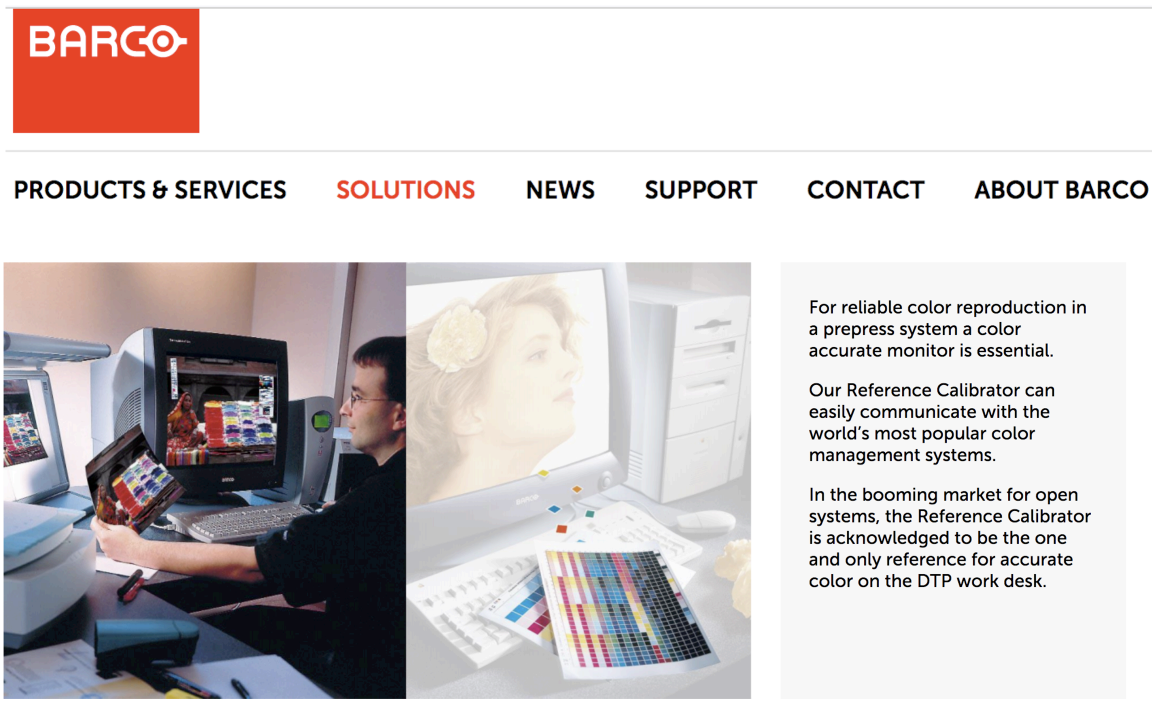
Due Wednesday, 9/29/21 at 11:59 PM

ECE 638 **Homework No. 3** **Fall 2021**

Be sure to turn in all your Matlab or Python code for the problems below.

1. This problem concerns calibration of a CRT monitor. From the course website, download the following files: *README.pdf*, *RGB_ch_xyY.txt*, *barco_125_xyY.txt*, and *barco_729_xyY.txt*. (These files can be found under “Data Files/barco”. Then, click “Index.html” to get a convenient links to the README file, as well as the data files.) “Check *README.pdf* first; so you will understand the format of the data. For the purposes of this homework problem, you may ignore the cautionary notes about low luminance.

   Barco is a company that manufactures high-end display devices and other imaging hardware. The image on the left below shows a picture of a CRT monitor like that from which the data that you will be using to solve this problem was acquired.



   a. Use the file *RGB_ch_xyY.txt* to determine the gray balance curves for each of the R, G, and B channels. You can use the gain-offset-gamma (GOG) model described on p. 10 of the reference “CRT Calibration Techniques for Better Accuracy Including Low Luminance Colors,” that can also be downloaded from the course website. (This file can be found under “References/Display”. The file name starts with “061106_Arslan”.) There are a variety of ways to determine the best-fitting parameters for the three channels. You can use a Matlab or Python optimization toolbox. See the reference above for specific suggestions. Provide the parameter values for your three GOG functions, and turn in plots of the data and the GOG curves fitted to the data.

   b. Using your GOG curves determined in part a.) above, determine the linear *RGB* values for the two datasets, *barco_125_xyY.txt* and *barco_729_xyY.txt*. Then using *barco_729_xyY.txt* as training data, perform a linear regression to find the best 3×3 transformation $\mathbf{T}$ from CIE XYZ to the linear RGB space of the monitor. Using *barco_125_xyY.txt* as the test set, determine the mean and maximum CIE $\Delta E_{a^*b^*}$ errors. To compute the $\Delta E_{a^*b^*}$ errors, you will need to transform to CIE L*a*b*, which will require that you choose a white point. You can use D50, for which (X,Y,Z) = (96.42, 100, 82.49).

Note: For this problem, you do not need to worry about setting the white point of the monitor.

2. One morning while we were getting ready to prepare breakfast, my wife Cecilia was sitting on the sofa in our living room. She exclaimed that of the four blue dishes stacked on our glass-topped coffee table, one was a different color (see Fig. 1(a) below), and that she had never noticed that before. I got to wondering how

(a) (b)

Fig. 1: Images for Problem 2.

different were these colors in terms of a uniform color space, such as L*a*b*. Since it would not be practical to cart our PhotoResearch 704 Spectraradiometer the 3,000 mile distance between my lab at Purdue and our home in Portland, OR, I thought that perhaps we could estimate the color difference in the following manner:

i. Assume that image shown in Fig. 1(a) is in the standard color space sRGB.

ii. Using known equations that can be found on the web, convert the image to CIE XYZ. (Don't forget to linearize, i.e. un-gamma correct.) See Problem 3(a) below for more details on this transformation.

iii. For each of the two regions indicated by the red boxes, extract its pixels, and obtain an average CIE XYZ value for all the pixels in each block.

iv. Convert these two numbers to CIE L*a*b*, and compute their difference in 1976 delta $E_{a^*b^*}$. You may again assume D50 for your illuminant.

In the mean time as I was thinking about this problem, the lighting in the living room changed since the sun had continued to rise. So I took another picture (Fig. 1(b)).

Download these two images from the course website; and complete the above four steps for Figs. 1(a) and 1(b). Compare the color differences that you obtain for the two images. Be sure to document the equations that you use to transform from sRGB to CIE XYZ. Include a print-out of each image showing exactly where you extracted the pixel values.

3. In class, we discussed the fact that each color imaging device (capture or output) has a certain gamut of colors associated with it, which can either be captured or

reproduced. In this problem, you will compare the gamut for a standard input device (sRGB) to that of the Indigo press, two models of which are shown in Fig. 2 below. In order to meaningfully compare gamuts, we need to work in a uniform color space. We will use the 1976 CIE L*a*b* space with a D50 white point.

a. Consider first the gamut of the sRGB color space. By performing gamma uncorrection (de-gamma) to obtain the linear sRGB coordinates and then applying a 3x3 matrix transformation, we can transform to CIE XYZ, then use standard equations to transform from CIE XYZ to CIE L*a*b*. You may assume a gamma of 2.2. The matrix below can be used to convert from linear sRGB coordinates to CIE XYZ with an assumed D50 white point. Note that while the definition of CIE XYZ does not incorporate a white point, the sRGB color space is defined with an assumed white point. This matrix is taken from the International Color Consortium (ICC) website (http://www.color.org/srgb04.xalter). Thanks to Alty Jumabayeva for finding it for me. I have posted the complete document (sRGB.pdf) from which this matrix is taken in the References section at the course website under the heading "transformation_between_color_spaces". (All that you need to solve this problem is the matrix below. You do not need to look at the ICC document.)

$$\begin{bmatrix} X_{D50} \\ Y_{D50} \\ Z_{D50} \end{bmatrix} = \begin{bmatrix} 0.436030342570117 & 0.385101860087134 & 0.143067806654203 \\ 0.222438466210245 & 0.716942745571917 & 0.060618777416563 \\ 0.013897440074263 & 0.097076381494207 & 0.713926257896652 \end{bmatrix} \begin{bmatrix} R_L \\ G_L \\ B_L \end{bmatrix}$$

b. If you transform each vertex of the sRGB color cube to CIE L*a*b*, plot these vertices in the L*a*b* color space, and then connect the vertices corresponding to the edges of the cube, you will get an estimate of what the sRGB gamut looks like in the L*a*b* color space. The vertices in the sRGB color space are given by the coordinates (R, G, B), where each of values R, G, B is either 0 or 255. This gamut representation implicitly assumes that the mapping from (gamma-corrected) sRGB to CIE L*a*b* is linear, which is not true. You can obtain a more accurate representation of the gamut if you sample each face of the sRGB color cube on a grid of points, transform each sample point to L*a*b*, and then plot these points in the L*a*b* color space.

c. Now consider the gamut of the CMYK input color space for the Indigo press. You can download from the course website a set of measurements of the L*a*b* coordinates corresponding to samples from the CMYK input color space for the Indigo press. The file is called "IT8_7_4_Meas_Gloss.txt". (It can be accessed by clicking on the "Data Files" link.) It is in the standard format for data measured by an X-Rite instrument, in this case, the *X-Rite Eye-One iSis*. It contains a header with information about the measurement conditions, as well as the format of the data that follows. The amounts of C, M, Y, and K are given as percentages between 0 and 100, rather than unsigned 8-bit characters. The figure below shows a picture of this instrument.

To simplify matters, and to make the computations more analogous to those described above for sRGB, let us assume that we will only use three colorants C, M, and Y. In this case, we achieve black, or shades of gray, by printing equal amounts of C, M, and Y, rather than using the K colorant. This print mode is called *Enhanced Productivity Mode* or *EPM*, since it reduces the cost of ink, and also enables faster printing, due to the particular architecture of the Indigo press. It is actually a popular print mode with users of the Indigo press. In terms of the file of L*a*b* values corresponding to the CMYK input color space, this means that we only consider sample points where K = 0. The file also includes spectral data, which we will not be using for this problem.

Now, as we did with the sRGB color space, we can first obtain an approximate representation of the gamut of the CMY color space of the Indigo press by only transforming the vertices of the CMY color cube to L*a*b*, and then connecting those vertices corresponding to edges of the cube. Again, this representation assumes a linear transformation from Indigo CMY to L*a*b*, which is not accurate. To obtain a more accurate representation of the gamut, we should plot the L*a*b* coordinates of the points corresponding to the faces of the CMY Indigo color cube. Note that this particular gamut **is** dependent on the particular halftoning algorithm that was used to render the specific CMY color coordinates.

d. Now, what do I actually want you to do for this homework problem?

   i. Plot both sRGB and CMY Indigo gamuts in L*a*b* based on the simplified linear assumption discussed above.

   ii. Plot more accurate sRGB and CMY Indigo gamuts in L*a*b* by transforming sample points on faces of the sRGB or CMY Indigo color cubes in L*a*b*. Show both on the same axes in L*a*b* color space.

   iii. Generate a different representation of these gamuts by plotting slices for constant L*. (The axes of the plots will be b* vs. a*.) You might show plots for L* = 20, 40, 60, and 80. (Show both sRGB and CMY Indigo gamut slices on the same axes.

   iv. DISCUSS your results! Generally speaking, how do the gamuts compare? What are the hue angles at which large chroma values (c* = sqrt(a*^2 + b*^2) can be achieved with each color system? Why? Considering the gamut slices for constant L* values how does the gamut area change as L* increases from very low values through the mid-values to very high values?

Fig. 2(a) HP Indigo Press 3050, which can print 2,000 4-color sheets/hr.

Fig. 2(b) HP Indigo Press 30000, which can print 4600 3-color sheets/hr.