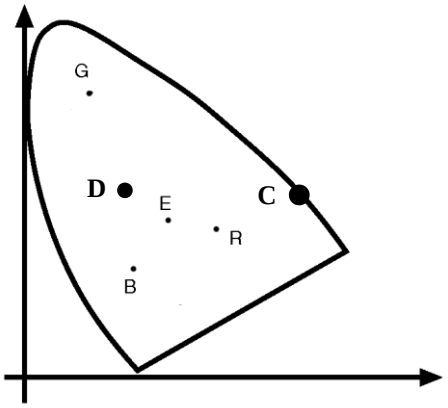
CS 576 – Assignment 2
Instructor: Parag Havaldar

Assigned on Mon 09/27/21,
Solutions due on Monday 10/11/21 by 12:00 noon

Theory Part (30 points)

Question1: Color Theory – 10 points
One of uses of chromaticity diagrams is to find the gamut of colors given the primaries. It can also be used to find dominant and complementary colors –
Dominant color of a given color D (or dominant wavelength in a color D) is defined as the spectral color which can be mixed with white light in order to reproduce the desired D color. *Complementary colors* are those which when mixed in some proportion create the color white. Using these definitions and the understanding of the chromaticity diagram that you have, answer the following.



i. In the image alongside find the dominant wavelength of color D. Show this wavelength. *(2 points)*
ii. Do all colors have a dominant wavelength? Explain your reasoning. *(3 points)*
iii. Find the color which is complimentary to the color C. Show this color *(2 points)*
iv. What colors in RGB color space map to the equiluminous point E upon projection into the chromaticity space. *(3 points)*

Question 2: Entropy Coding *(10 points)*
Consider a communication system that gives out only two symbols X and Y. Assume that the parameterization followed by the probabilities are $P(X) = x^k$ and $P(Y) = (1-x^k)$

- Write down the entropy function and plot it as a function of x for k=2. (*1 points)*
- From your plot, for what value of x with k=2 does H become a minimum? (*1 points)*
- Your plot visually gives you the minimum value of x for k=2, find out a generalized formula for x in terms of k for which H is a minimum *(3 points).*

- From your plot, for what value of x with k=2 does H become a maximum? (*1 points)*
- Your plot visually gives you the maximum value of x for k=2, find out a generalized formula for x in terms of k for which H is a maximum *(4 points)*.

Question 3: Huffman Coding *(10 points)*
Consider a source that emits an alphabet with three symbols A, B, C having probability distributions as follows - P(A) = 0.625, P(B) = 0.125, P(C) = 0.25

- Construct a Huffman code and calculate its average code length. (*1 Point*)
- For this three-symbol case, how many Huffman codes are possible. What are they? (*1 Point*)
- In general, for N symbols, how many Huffman codes are possible? Justify your answer formally. (*3 Points*)
- Is the code you have defined optimal - give reasons! If not, what can you do to improve upon this? Show a solution giving your codes and compute the improved avg. code length. (*5 Points*)

Programming Part (*70 points*)

This assignment will help you gain a practical understanding of analyzing color channels especially as it pertains to chroma keying. In this programming assignment, you will write a program to composite two images with an implicit alpha channel. When compositing images, the foreground objects are normally captured in front of a "chroma" colored background – such as blue, green, black etc. During compositing, whatever is not the background chroma, will be a foreground object and is composited with the background object using relations discussed in class. Write a program which will take the foreground object and background image as input arguments and produce a composited image. Specifically you will have to

- Given a foreground image, find out what is the "chroma" of the image. This can be done by statistically sampling the colors of the image and creating a histogram. Assume that pixels which correspond to the chroma have the maximum frequency. Be aware that due to noise and quantization effects, this chroma color may not be a specific color but may be colors lying in a small range.
- Once the chroma has been detected, you can now remove the pixels which correspond to the object in the foreground images and composite them into the background image.
- Sometimes you will notice that aspects of the foreground image won't blend well into the background in certain high frequency areas. These are due to aliasing. Your program should appropriately remove such effects.

Images will be placed on the website and the format explained, which will be used for display. Assume that both foreground and background images will be of the same size. We should be able to run compile and run your code as

"yourProgram foregroundImage backgroundImage"

which should display an composited output.

Some thoughts that you might want to consider

- The foregroundImage has pixels with background samples (that need to be replaced) and foreground samples (that need to be composited correctly). The background samples here may be of any color (not necessarily green), but statistically will have very close values because of shadows, lighting variations etc (see image below). How do you detect the thresholds to label a pixel as a background pixel given that the screen pixels might have noise, shadows etc.

- You can certainly make this work by processing in the RGB color space by computing appropriate statistical thresholds, but other color spaces might work better like HSV. Here are references that were shared in the class and should give you an understanding of these spaces.
  https://en.wikipedia.org/wiki/HSL_and_HSV
  https://www.cs.rit.edu/~ncs/color/
- To have good quality at the boundaries of the composition (where foreground regions and background video meet), can you think how you to blend boundary pixels correctly, so that the foreground element does not appear stuck on a background but blends in nicely

What should you submit?

- Your source code, and your project file or makefile, if any, using the submit program. ***Please do not submit any binaries or images***. We will compile your program and execute our tests accordingly.
- If you need to include a readme.txt file with any special instructions on compilation, that is fine too.