Digital Image Processing Laboratory: Introduction to Colorimetry

December 21, 1998

1 Introduction

In order to understand natural images, one needs to understand how objects reflect light and how humans perceive that reflected light as color. The objective of this laboratory is to introduce the basic ideas behind colorimetry, the quantitative measurement and manipulation of color. After you have completed this laboratory, you will understand how to display images on a calibrated monitor to accurately produce the colors of an image.

There are three major components to understanding color:

- Understanding how light is reflected from a natural scene.
- Understanding how humans perceive reflected light.
- Understanding how a rendering device, such as a monitor, functions.

The following sections of this introduction will introduce the theoretical background for each of these components, and then the laboratory will ask you to use this theory to perform specific tasks.

In section 3, we will reproduce a chromaticity diagram from the color matching functions. Within this diagram, we will plot two standard sets of RGB primaries and two white points. In section 4, we will reproduce an image given a set of reflectance coefficients, the radiant energy from an illuminant, and the color matching functions. Finally in section 5, we will create a color-filled chromaticity diagram using the set of Rec. 709 RGB primaries and a D_{65} white point. Each exercise will be performed in Matlab.

1.1 Reflection of Light in a Natural Scene

It is important to note that the color that one sees is a function of both the reflectance of an object's surface, and the spectral distribution of the light source, or *illuminant*. More specifically, let $S(\lambda)$ be the spectral energy density of an illuminant as a function of the spectral wavelength λ, and let $R(\lambda)$ be the fraction of the illuminant energy that is reflected toward the viewer. The spectral distribution of light that the viewer sees is given by

$$I(\lambda) = R(\lambda)S(\lambda) \ . \tag{1}$$

Questions or comments concerning this laboratory should be directed to Prof. Charles A. Bouman, School of Electrical and Computer Engineering, Purdue University, West Lafayette IN 47907; (765) 494-0340; bouman@ecn.purdue.edu

The viewer's perception of color depends on the distribution of the reflected light $I(\lambda)$. While the spectral distribution of light energy in a natural scene can be very complex, human beings are only sensitive to a limited range of spectral wavelengths. We will be principally concerned with wavelengths between 400 and 700 nanometers (nm) since this is the range of wavelengths that are most visible. Wavelengths longer then 700nm are known as infrared, and those shorter then 400nm are known as ultraviolet.

Importantly, $I(\lambda)$ is a function of both the object being viewed and the illuminant $S(\lambda)$. In practice, we this means that the perceived color of an object can depend on the spectral distribution of light which is illuminating it. Illuminants can vary widely in their spectral distribution, but there are some standard models that are often used to model illuminants. One simple model is the equal energy white illuminant given by

$$S_{EE}(\lambda) = 1 \ .$$

This illuminant has equal spectral energy at each wavelength. In practice few illuminants have such flat spectral distributions. A more common model for an illuminant is the standard D_{65} illuminant. This illuminant approximates the color of sunlight and is given by a tabulated set of measurements which we will use in this laboratory. For this laboratory, all spectral distributions will be given at 31 discrete wavelengths which are uniformly spaced between 400 and 700 nanometers (nm), i.e. $\lambda = 400nm, 410nm, \ldots, 700nm$.

1.2 Human Perception of Color

Humans have three distinct types of color receptors or cones in their retinas. The long, medium and short cones are sensitive to long, medium and short wavelengths of light. The response of each cone to light has been measured as a function of wavelength λ and is represented by the three functions $l_0(\lambda)$, $m_0(\lambda)$, and $s_0(\lambda)$. These function are referred to as the color matching functions. The total response of each cone is then given by

$$\begin{aligned} L &= \int_\lambda I(\lambda) l_0(\lambda) d\lambda \qquad (2) \\ M &= \int_\lambda I(\lambda) m_0(\lambda) d\lambda \\ S &= \int_\lambda I(\lambda) s_0(\lambda) d\lambda \end{aligned}$$

where L, M, and S are the responses of the long, medium, and short cones respectively. Notice that while the spectrum of incoming light may be very complex, the response of the visual system is limited to three dimensions. This three dimensional representation of color is known as the *tristimulus* model of color vision and is widely accepted and known to be quite accurate.

In practice, these integrals are approximated by summing over a finite number of wavelengths. Typically, the spectral energy is measured at 31 discrete wavelengths which are uniformly spaced between 400 and 700 nanometers (nm), $\lambda = 400nm, 410nm, \ldots, 700nm$. The tristimulus values of (2) are then approximated by three vector inner products.

$$L = l_0 I \qquad (3)$$

$$M = m_0 I$$
$$S = s_0 I$$

where l_0, m_0, and s_0 are 1×31 row vectors containing properly normalized samples of the color matching functions, and I is a 31×1 column vector representing the spectral energy of the incoming light. This relationship my be more compactly expressed in matrix form.

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} l_0 \\ m_0 \\ s_0 \end{bmatrix} I$$

In fact, the LMS system is not commonly used to specify colors. The most standard method for specifying tristimulus values is with the CIE 1931 standard XYZ color coordinates. The XYZ system is related the LMS system through the following transformation.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = A \begin{bmatrix} L \\ M \\ S \end{bmatrix}$$

The color matching functions x_0, y_0, and z_0 are then related to l_0, m_0, and s_0 by the same transformation.

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = A \begin{bmatrix} l_0 \\ m_0 \\ s_0 \end{bmatrix}$$

The transformation matrix is approximately given by

$$A^{-1} = \begin{bmatrix} 0.2430 & 0.8560 & -0.0440 \\ -0.3910 & 1.1650 & 0.0870 \\ 0.0100 & -0.0080 & 0.5630 \end{bmatrix}$$

In this case, the XYZ coordinates are computed via the relation

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} I$$

The XYZ color matching functions x_0, y_0, and z_0 are chosen to have some important special properties. First, they are positive for all λ. Second, for an equal energy spectral distribution with $I_{EE} = [1, 1, \cdots, 1]^t$, the XYZ coordinates have unit values.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} I_{EE} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

A set of tristimulus values (X, Y, Z) can be thought of as a vector in a 3-D color space. The color of a point in the space is then dependent on both the direction and magnitude of the vector. Intuitively, changes in the vector's magnitude will change the brightness of the color while changes in the vector's direction will change the color's hue and saturation. These latter properties are generally referred to as a color's *chromaticity*. The chromaticity may be computed by normalizing the XYZ components by their sum.

$$\begin{aligned} x &= \frac{X}{X+Y+Z} \\ y &= \frac{Y}{X+Y+Z} \\ z &= \frac{Z}{X+Y+Z} \end{aligned} \tag{4}$$

Geometrically, the chromaticity of a color is the intersection of the vector (X, Y, Z) with the unit plane $X + Y + Z = 1$. Notice from (4) that the chromaticities are independent of the length of the (X, Y, Z) vector, and that $x + y + z = 1$. Often, only (x, y) is specified for a color's chromaticity since z may be computed by $z = 1 - x - y$.

1.3 Color Rendering on a CRT Monitor

The XYZ color coordinates precisely specify what a color should look like. However, in most practical applications, we still need a way to accurately produce those colors on a display or other rendering device. In this section, we will explain how a color can be transform from XYZ to the color coordinates of a typical CRT monitor.

1.3.1 Color Primary Transformations

Most display devices produce colors by linearly adding together three primary colors such as red, green, and blue. These three primary colors may be represented by three basis vectors $\vec{R}$, $\vec{G}$, and $\vec{B}$. Using this convention, a color in RGB may be represented as

$$r\,\vec{R} + g\,\vec{G} + b\,\vec{B} = \begin{bmatrix} \vec{R} & \vec{G} & \vec{B} \end{bmatrix} \begin{bmatrix} r \\ g \\ b \end{bmatrix} . \tag{5}$$

We may specify the three RGB primaries by measuring their XYZ components

$$\vec{R} = \begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix} \quad \vec{G} = \begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix} \quad \vec{B} = \begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix}$$

where the subscripts r, g, and b specify the XYZ measurements for each of the three primaries. Combining this with equation (5), the XYZ values of a color (r, g, b) can be written as the matrix product

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = M \begin{bmatrix} r \\ g \\ b \end{bmatrix} \tag{6}$$

where

$$M = \begin{bmatrix} X_r & X_g & X_b \\ Y_r & Y_g & Y_b \\ Z_r & Z_g & Z_b \end{bmatrix} . \tag{7}$$

Similarly, the (r, g, b) values may be computed from XYZ via

$$\begin{bmatrix} r \\ g \\ b \end{bmatrix} = M^{-1} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} .$$

1.3.2 White Point

Normally, the color primaries $\vec{R}$, $\vec{G}$, and $\vec{B}$ are specified indirectly through their chromaticities and white point. For example, the CIE 1931 standard RGB primaries have the chromaticity values given below.

$$\begin{aligned} \vec{R} &= (x_r, y_r, z_r) = (0.73467, 0.26533, 0.0) \\ \vec{G} &= (x_g, y_g, z_g) = (0.27376, 0.71741, 0.00883) \\ \vec{B} &= (x_b, y_b, z_b) = (0.16658, 0.00886, 0.82456) \end{aligned} \tag{8}$$

These primaries are very useful for laboratory measurements, but are not practical for use in devices such as monitors. A more common choice of primaries for a display device are the recommended standard 709 RGB primaries given below.

$$\begin{aligned} \vec{R}_{709} &= (x_r, y_r, z_r) = (0.640, 0.330, 0.030) \\ \vec{G}_{709} &= (x_g, y_g, z_g) = (0.300, 0.600, 0.100) \\ \vec{B}_{709} &= (x_b, y_b, z_b) = (0.150, 0.060, 0.790) \end{aligned} \tag{9}$$

Remember that chromaticities are normalized such that $x + y + z = 1$, so they do not completely specify the XYZ coordinates required to compute the matrix M of equation (7). In general, the XYZ coordinates are related to the chromaticities through three constants κ_r, κ_g, and κ_b.

$$\begin{aligned} (X_r, Y_r, Z_r) &= \kappa_r(x_r, y_r, z_r) \\ (X_g, Y_g, Z_g) &= \kappa_g(x_g, y_g, z_g) \\ (X_b, Y_b, Z_b) &= \kappa_b(x_b, y_b, z_b) \end{aligned}$$

Using this notation

$$M = \begin{bmatrix} x_r & x_g & x_b \\ y_r & y_g & y_b \\ z_r & z_g & z_b \end{bmatrix} \begin{bmatrix} \kappa_r & 0 & 0 \\ 0 & \kappa_g & 0 \\ 0 & 0 & \kappa_b \end{bmatrix} \tag{10}$$

To uniquely define M, we will specify the *white point* of the primaries. The white point is the color produced by adding unit amounts of the three primaries, i.e. when $(r, g, b) =$

$(1,1,1)$. For a CRT monitor, this would be the shade of white produced by applying the maximum (r,g,b) inputs of $(255,255,255)$. Normally, the white point is specified using chromaticity components (x_{wp}, y_{wp}, z_{wp}). If we assume that the monitors brightest output is at $Y = 1$, then the XYZ coordinates are given by

$$\begin{bmatrix} X_{wp} \\ Y_{wp} \\ Z_{wp} \end{bmatrix} = \begin{bmatrix} x_{wp}/y_{wp} \\ 1 \\ z_{wp}/y_{wp} \end{bmatrix}$$

Then by applying equations (6) and (10), we have that

$$\begin{bmatrix} x_{wp}/y_{wp} \\ 1 \\ z_{wp}/y_{wp} \end{bmatrix} = \begin{bmatrix} x_r & x_g & x_b \\ y_r & y_g & y_b \\ z_r & z_g & z_b \end{bmatrix} \begin{bmatrix} \kappa_r & 0 & 0 \\ 0 & \kappa_g & 0 \\ 0 & 0 & \kappa_b \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{11}$$

so that

$$\begin{bmatrix} \kappa_r \\ \kappa_g \\ \kappa_b \end{bmatrix} = \begin{bmatrix} x_r & x_g & x_b \\ y_r & y_g & y_b \\ z_r & z_g & z_b \end{bmatrix}^{-1} \begin{bmatrix} x_{wp}/y_{wp} \\ 1 \\ z_{wp}/y_{wp} \end{bmatrix} \tag{12}$$

A commonly used white point is D_{65}.

$$x_{D65} = 0.3127;\ y_{D65} = 0.3290;\ z_{D65} = 0.3582 \tag{13}$$

The chromaticities of D_{65} are exactly those derived from the D_{65} illuminant.

2 Plotting Color Matching Functions and Illuminants

In this section, you will plot the color matching functions and illuminants used in the laboratory.

1. Load the file *data.mat* into Matlab. It contains the color matching functions x_0, y_0, and z_0. It also contains the spectral distribution of two illuminants. The `illum1` vector corresponds to a D_{65} source, and `illum2` corresponds to a fluorescent light source.

2. Plot the three x_0, y_0, and z_0 color matching functions.

3. Plot the three l_0, m_0, and s_0 color matching functions corresponding to the long medium and short cones.

4. Plot the spectrum of the D_{65} and fluorescent illuminants.

Section 2 Report:
Hand in:

- The plot of x_0, y_0, and z_0 color matching functions.
- The plot of l_0, m_0, and s_0 color matching functions.
- The plot of the D_{65} and fluorescent illuminants.

3 Chromaticity Diagrams

A chromaticity diagram is a graphical representation of colors according to their position in (x, y) chromaticity coordinates. Chromaticity coordinates have an important property that combinations of any two colors always fall along a straight line. This property will be useful in visualizing the structure of color space.

The most saturated colors are produced by spectral distributions that only have energy at a single wavelength. Such a pure spectrum can only be generated by a very narrow optical filter or a narrow band spectral source such as a laser. The XYZ coordinates of a pure spectral source at wavelength λ_0 is then given by

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \int_\lambda \delta(\lambda - \lambda_0) x_0(\lambda) d\lambda \\ \int_\lambda \delta(\lambda - \lambda_0) y_0(\lambda) d\lambda \\ \int_\lambda \delta(\lambda - \lambda_0) z_0(\lambda) d\lambda \end{bmatrix} = \begin{bmatrix} x_0(\lambda_0) \\ y_0(\lambda_0) \\ z_0(\lambda_0) \end{bmatrix} .$$

Since the chromaticities of these pure spectral colors are highly saturated, they represent a boundary for all the colors that can be generated. A plot of these (x, y) chromaticities as a function of λ is known as a chromaticity diagram.

1. Load the Matlab data file `data.mat` into your Matlab.
2. In the following tasks, you will be asked to plot multiple items on a single graph. You can do this by using the Matlab command `hold on`. When your plot is complete you may use the command `hold off`.
3. Using the equations in (4), plot the (x, y) chromaticities of a pure spectral source as a function of λ. Use a solid line type.
4. In your diagram, also plot the three CIE 1931 standard RGB primaries using the chromaticities given in equation (8). Connect the points of the three primaries with straight lines to show the range of colors that they can generate. Label each of these points using the `text` command.
5. Repeat the previous step using the Rec. 709 RGB primaries given in equation (9).
6. Plot and label a point on your diagram for the chromaticity of a equal energy white point.

7. Plot and label a point on your diagram for the chromaticity of a D_{65} white point given in equation (13).

8. Print out your final figure. Use the command `orient tall` to use the full page for your figure.

Section 3 Report:
Hand in your labeled chromaticity diagram.

4 Rendering an Image from Illuminant, Reflectance, and Color Matching Functions

The objective of this section will be to display a calibrated color image from a known D_{65} illuminant and the reflectances at each point in the image. As in the previous section, you will use the XYZ color matching functions and two illuminants, but in this section your will also use a $170 \times 256 \times 31$ array `R` which consists of a 2-D matrix of reflectance coefficients for each of the 31 wavelengths. The indexing of the Matlab vector has the order *R(row,column,wavelength)*. **Beware** this file it is about 11 Mbytes!

1. Load the *data.mat* and *reflect.mat* data files into Matlab.

2. Using the D_{65} illuminant, for each pixel compute the intensity of the reflected light energy at each given λ using equation (1). Call this $170 \times 256 \times 31$ array `I`.

3. Compute the tristimulus values in XYZ for each pixel by applying the color matching functions. Call this $170 \times 256 \times 3$ array `XYZ`.

4. Compute the transformation matrix M to convert from XYZ coordinates to the Rec. 709 RGB primaries with a D_{65} white point. Print out this matrix.

5. Use M to transform each pixel in your `XYZ` array into RGB coordinates.

6. Now γ correct your image so that it will be displayed properly on your monitor, and multiply the gamma corrected result by 255 so that it will have the proper range. Assume that your monitor has a γ of 2.2.

7. Write the image out to a TIFF file *image1.tif*. **Note** that you need to first convert it to type *uint8*, then use the Matlab function *imwrite*.

8. Repeat the entire exercise using the fluorescent source in `illum2`, and write out the result as *image2.tif*.

9. Compare the two images on your display, and print out the two images.

Section 4 Report:
Hand in the following:

1. A hard copy of the matrix M.
2. The print outs of the resultant images using D_{65} and fluorescent illuminants.
3. An description of the differences in the two images.

5 Color Chromaticity Diagram

Now we will create a color-filled chromaticity diagram.

1. Use `meshgrid` to create two matrices, x and y, of chromaticity values ranging from 0 to 0.9, with a spacing of 0.005. Then compute the corresponding z matrix.
2. For each position in the chromaticity matrices, compute the corresponding display (r, g, b) values assuming a monitor with:
   - Rec. 709 RGB primaries
   - D_{65} white point
   - $\gamma = 2.2$

   Place the three computed values in a 3-D color image array.
3. The locations where r, g, **or** b are negative represent colors that cannot be reproduced by this set of primaries. In each location where the r, g, or b components are negative, set all the components to zero (black).
4. The Matlab `image` command assumes that color images have components that lie between zero and one. Linearly rescale the entire image so that the maximum value of any component corresponds to one. In other words, divide the entire array by the maximum pixel value.
5. Use the Matlab command `image` to display your color chromaticity diagram. Label the x and y axes appropriately, and print your color diagram.

Section 5 Report:
Hand in your color diagram.