ACTIVITY 6 – ENHANCEMENT OF COLOR IMAGES

Introduction

A colored digital image is formed from the overlay of three matrices representing the redness, greenness, and blueness of an image. Reading a colored image from your scientific software you will see that the size of the image is MxNx3 where MxN is the size of the image, and 3 is because of the red, green, and blue channels of the image. Per pixel, the color captured by a digital color camera is an integral of the product of the spectral power distribution of the incident light source $M_e(\lambda)$, the surface reflectance $\rho(\lambda)$ and the spectral sensitivity of the camera $S_i(\lambda)$. That is, if the color camera has spectral sensitivity $S_R(\lambda)$, $S_G(\lambda)$, $S_B(\lambda)$ for the red, green and blue channels respectively, then the color of the pixel is given by

$$DN_R = K_R \sum M_e(\lambda)\rho(\lambda)S_R(\lambda) \qquad (1)$$

$$DN_G = K_G \sum M_e(\lambda)\rho(\lambda)S_G(\lambda) \qquad (2)$$

$$DN_B = K_B \sum M_e(\lambda)\rho(\lambda)S_B(\lambda) \qquad (3)$$

where

$$K_i = \frac{1}{\sum M_e(\lambda)S_i(\lambda)} \qquad (4)$$

is a balancing constant equal to the inverse of the camera output when shown a white object ($\rho(\lambda) = 1.0$). Equations (1) to (4) explain why sometimes the colors captured by a digital camera appear unsatisfactory.

White Balancing

If you inspect your digital camera you will find that it has a setting called “WHITE BALANCE” or “WB”. This setting allows the user to select the white balancing constants appropriate for the capturing conditions. For example, you may have seen icons such as a sun (for sunny, bright outdoors), cloud (for cloudy day), light bulb (for incandescent light) and a fluorescent lamp (for fluorescent lighting). You may also have seen “AWB” which stands for Automatic White Balancing, and symbols such as “6500K” which stands for the color temperature of the ambient light. By setting your camera white balance to the illumination condition, you will obtain an image where white appears white and the rest of the colors are properly rendered.

White balancing algorithms can be used to restore faded color photographs or correct unbalanced images. How do you know if an image is unbalanced? If there is a known white object in the scene but it does not look white, it is unbalanced. Skin color is another indicator. If there are people in the picture and their skin color looks purple, bluish or even greenish (and you're certain they're not sick or dead- haha) then the image may be unbalanced.

Automatic White Balancing Algorithms

There are three popular algorithms for achieving automatic white balance. These are **Contrast Stretching**, **White Patch Algorithm,** and the **Gray World Algorithm**.

Contrast Stretching in RGB

In Activity 5 we contrast stretched the histogram of a grayscale image. The same can be done for color images but the contrast stretching is done per channel.

White Patch Algorithm

If you look carefully at equations (1) to (3) these are just the raw camera outputs divided by the camera output for a white object. So the key is to determine what the camera output is for a white object. In the White Patch Algorithm use the RGB values of a known white object as the denominator in Equation 4 .

Gray World Algorithm

In the Gray World Algorithm, it is assumed that the average color of the world is gray. Gray is part of the family of white, as is black. Therefore, if you know the RGB of a gray object, it is essentially the RGB of white times a constant factor. Thus, to get the balancing constants, you take the average red, green and blue value of the captured image and use them as the denominator in Equation 4.

Procedure

1. Can white balancing restore faded photographs? Scan a faded or unbalanced photograph or get one from your collection. Note: photograph must originally be in color. Sepia photos do not count as colored photographs.
2. Apply all three white balancing algorithms to your image.
   a. Contrast Stretching
      i. If you load the image as I let R = I(:,:,1); G = Im (:,:,2); B = (:,:,3).
      ii. If you imshow these channels they look gray. The graylevel indicates how bright the red, green or blue light must be to reconstruct the color.
      iii. Perform contrast stretching on each of these channels and overlay them into one matrix , e.g, I_restored (:,:,1) = R_stretched, I_restored(:,:,2) = G_stretched, I_restored(:,:,3) = B_stretched, and then imshow I_restored.
   b. Gray World Algorithm
      i. Get the average value of the red, green and blue channel of your image, Rave, Gave and Bave, respectively.
      ii. To get the white balanced image, divide each RGB channel by the respective averages, that is , Rwb = R/Rave, Gwb = G/Gave, Bwb=B/Bave. Notice this is the same gist as Equations 1 to 4. Display the image.
   c. White Patch Algorithm. This technique only works if there are white parts in the image. Skip this if there are none.

i. Get a region from the image which is known to be white (e.g. white sando or uniform, white sand, white paper, etc.) Average the RGB of the white pixels to get Rw, Gw, Bw, separately.
ii. Divide each channel with the whole original image with the respective white averages from each channel, that is, Rwb = R/Rw, Gwb=G/Gw, Bwb = B/Bw. Display the image.

3. Comment on how each algorithm fared in restoring your image.