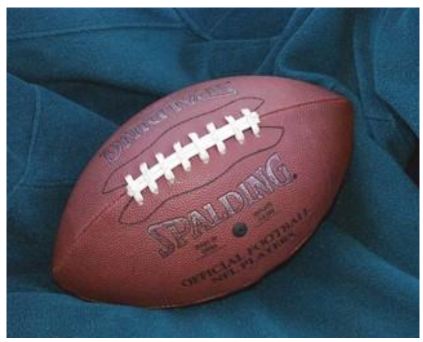
Experiment 5: Color Image Processing

1. Basics of colour image processing

Objective: To understand colour images

Theory: Colours are powerful descriptors that often simplifies object identification and extraction from a scene. The human eye has only three different colour receptors, each of them sensible to one of the three primary colours red, green and blue. When photons hit the retina of eye, they give rise to electric impulses, which on reaching the brain, are translated into colour. Different wavelengths of light are perceived as different colours. Humans can discern thousands of colour shades and intensities, compared to about only two dozen shades of gray. Digital images that include color information for each pixel are called colour images.

Figure shown below is a sample color image available in Matlab.



MCQ:

1. When an RGB image is displayed, the entire image is black. Then,
   a) each pixel in the image have the same, but any nonzero value
   b) all the three colour components have different values
   c) elements in each colour components have same value
   d) all the pixels have the same value of zero.

2. Suppose X, Y and Z are three matrices of size 256x256 and X and Y have the same elements. Assume that all the elements in Z has a value of 255 and Y is a zero matrix. If X,Y and Z corresponds to R,G and B components of an image A displayed,
   a) The image A is Black
   b) The image A is Blue
   c) The image A is Yellow
   d) The image A is Red

Matlab Assignment:

1. Write a Matlab program to generate a grayscale image from a colour image.
2. Write a Matlab program to read an RGB image and extract the three colour components, red, green, blue.
   Note: The available colour images in Matlab includes: football.jpg, hestain.png, peppers.png

Steps to generate a grayscale image from colour image:

i) Read a colour image. (Use: imread)
ii) Convert to gray scale image (Use: rgb2gray)
iii) Display original and converted image.(Use: imshow)

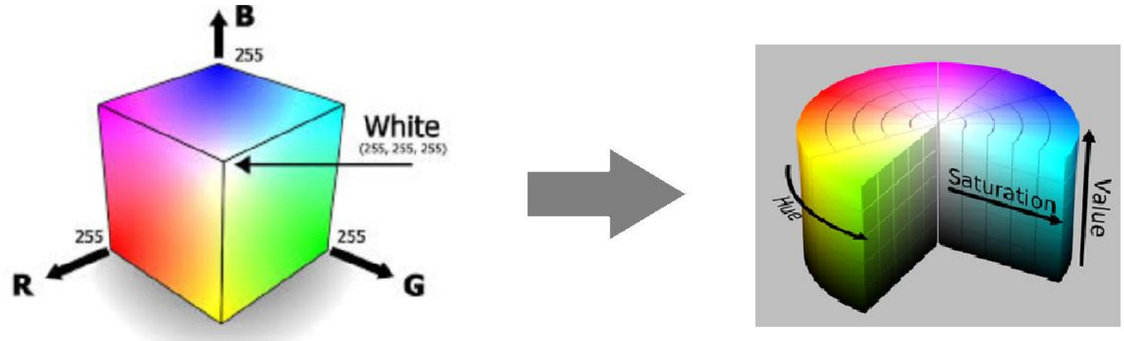
Steps to extract the three colour components from colour image:

i) Read a colour image A. (Use: imread)
ii) Make three copies of A as R,G and B (Note: each of R,G and B matrices have separate red, green and blue components as in A)
iii) To get the red component, Make green and blue components of R as zero matrices
iv) To get the green component, Make red and blue components of G as zero matrices
v) To get the blue component, Make red and green components of B as zero matrices
vi) Display A,R,B and G.(Use: imshow, subplot)

2. Colour model conversion

Objective: To learn RGB to HSV conversion.

Theory: Colour models provide a standard way to specify a particular colour by defining a 3D coordinate system, and a subspace that contains all constructible colours within a particular mocel. Any colour can be specified using a model. In the RGB colour model, the three primary colours red, green and blue form the axis of a cube. This model is based on a Cartesian coordinate system. HSV stands for hue, saturation, and value and is a cylindrical-coordinate representations of points in an RGB color model. In the cylinder, the angle around the central vertical axis corresponds to "hue", the distance from the axis corresponds to "saturation", and the distance along the axis corresponds to "value". So, it is important to convert RGB to HSV for separating the information such as brightness, pureness and colour.



RGB to HSV conversion formula:

- First, the *R*,*G*,*B* values are divided by 255 to change the range from [0 255] to [0 1].

  R' = R/255
  G' = G/255
  B' = B/255
  C_{max} = max(R', G', B')
  C_{min} = min(R', G', B')
  $\Delta = C_{max} - C_{min}$

- Hue calculation

$$H = \begin{cases} 0^0 & ,\Delta = 0 \\ 60^0 \times \left(\frac{G' - B'}{\Delta} mod6\right) & ,C_{max} = R' \\ 60^0 \times \left(\frac{B' - R'}{\Delta} + 2\right) & ,C_{max} = G' \\ 60^0 \times \left(\frac{R' - G'}{\Delta} + 4\right) & ,C_{max} = B' \end{cases}$$

- Saturation calculation

$$S = \begin{cases} 0 & ,C_{max} = 0 \\ \frac{\Delta}{C_{max}} & ,C_{max} \neq 0 \end{cases}$$

- Value calculation

  V= C_{max}

MCQ:

1. ______ is the main indicator of colour.

a) Hue
b) Saturation
c) Value
d) Intensity

2. If each element in the R,G and B component matrices are represented with 8-bit numbers, the total number of colours in the image is:
   a) 2^8
   b) $2^8 + 2^8 + 2^8$
   c) $2^8 \times 2^8 \times 2^8$
   d) 3×2^8

Matlab Assignment:

1. Write a Matlab program to display the histogram of the saturation component of a colour image.

Steps to display the histogram of the saturation component:

i) Read a colour image A. (Use: imread)
ii) Convert the format from RGB to HSV (Use: rgb2hsv)
iii) Separate the saturation component
iv) Compute and display the histogram of saturation component (Use: imhist)

3. Color image enhancement

Objective: To understand enhancement in color images

Theory: Image Enhancement basically improves the visual quality of an image by providing clear images. Based on color, images can be classified as gray scale and true color images. True color images represent full range of available colors that are similar to actual object. However a color images occupy more space when compared to gray scale images, they are very much useful for many applications. Every pixel of color image has both color and intensity. For visually acceptable results, it is necessary and almost sufficient to provide three color channels for each pixel. Color images can be represented by a stack of three matrices.

In order to enhance such images first the RGB images are converted to HSI format. After enhancing different layers, inverse transformation is applied to get the enhanced images.

MCQ:

1. Full form of HSI format is

   a) Hue Stain Intensity (b) High Saturation Intensity (c) Hue Saturation Intensity
   (d) High Saturation Index

2. HSV and HSI color format are same or not?
   a) N0 b) YES

3. Which one of the following is not a color format?
   a) YCbCr b) CMYK (c) YCbPr (d) CMYC

Example of color image enhancement:

Low Contrast image

After enhancement

Assignment:

1. Write a program to perform following enhancement operations on a color image
   (i) Point processing (Image negative, thresholding, contrast stretching)
   (ii) Spatial filtering(LPF and HPF)
   (iii) Histogram Processing

Steps:

1. Convert the RGB image HSV format (Previous assignment)
2. Enhance it (Either point, spatial or histogram) – Experiment 2 and 3
3. Convert back to RGB