# EXPERIMENT – 8

## Aim:

Perform various colour transformations on an image.

Software used: MATLAB

## Theory:

Colour can be described by its red (R), green (G) and blue (B) coordinates (the well-known RGB system), or by some its linear transformation as XYZ, CMY, YUV, IQ, among others. The CIE adopted systems CIELAB and CIELUV, in which, to a good approximation, equal changes in the coordinates result in equal changes in perception of the colour. Nevertheless, sometimes it is useful to describe the colours in an image by some type of cylindrical-like coordinate system, it means by its hue, saturation and some value representing brightness. If the RGB coordinates are in the interval from 0 to 1, each colour can be represented by the point in the cube in the RGB space. Let us imagine the attitude of the cube, where the body diagonal linking” black” vertex and” white” vertex is vertical. Then the height of each point in the cube corresponds to the brightness of the colour, the angle or azimuth corresponds to the hue and the relative distance from the vertical diagonal corresponds to the saturation of the colour.

**Code:**

RGB = imread('MATLAB Drive/927310.jpg');

subplot(2,3,1);

imshow(RGB);

title('Original image')

% 1

HSV = rgb2hsv(RGB);

[h,s,v] = imsplit(HSV);

saturationFactor = 2;

s\_sat = s\*saturationFactor;

HSV\_sat = cat(3,h,s\_sat,v);

RGB\_sat = hsv2rgb(HSV\_sat);

subplot(2,3,2);

imshow(RGB\_sat)

title('HSV to RGB image')

% 2

RGB = reshape(ones(64,1)\*reshape(jet(64),1,192),[64,64,3]);

HSV = rgb2hsv(RGB);

[h,s,v] = imsplit(HSV); %Split the HSV version of the synthetic image into its component planes:

%hue, saturation, and value.

subplot(2,3,3);

montage({h,s,v,RGB},"BorderSize",10,"BackgroundColor",'w'); %Display the individual HSV color

%planes with the original image

%Split RGB Image into Its Component Channels

I = imread('MATLAB Drive/927310.jpg');

imshow(I)

[r,g,b] = imsplit(I);

subplot(2,3,4);

montage({r,g,b},'Size',[1 3])

%Split Image in HSV Colorspace into Its Component Channels

rgbImage = imread('MATLAB Drive/927310.jpg');

imshow(rgbImage)

hsvImage = rgb2hsv(rgbImage);

[h,s,v] = imsplit(hsvImage);

subplot(2,3,5);

title('final image');

montage({h,s,v},'Size',[1 3])

**Output:**

Graphical user interface

Description automatically generated with medium confidence