**PROJECT-5**

**COLOR TRANSFORMATION**

**PART - 3**

EE5356 Digital Image Processing

Dr. K. R. Rao

Submitted By:

Ashutosh Desai

UTA ID:1001388602

Email: - ashutoshrajesh.desai@mavs.uta.edu

March 6th 2018

**EE5356 LAB Assignment #3B**

The test image, Girl256color.raw.

**1) From project #3, use the RGB color components to obtain Y,I,Q.**

Y: luminance, I: hue, Q: saturation



Show the luminance Y and chrominance I and Q components.

**2) Use the matrix transformation**



Show the R,G,B components.

See ‘Color Space and Residual Color Transform Support’ section in the paper G. Sullivan, P. Topiwala and A.Luthra “The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions”, SPIE Conference on Applications of Digital Image Processing XXVII Special Session on Advances in the New Emerging Standard:H.264/AVC, vol. 5558, pp.53-74, Aug. 2004.

**3) Perform color space conversion from R,G, B to Y,Cg,Co given flowers.bmp (500x362) on the course website (Cg = Green Chroma, Co=Orange Chroma)**

Show the images as Y,Cg,Co components

**4) Perform the inverse color space conversion from Y,Cg,Co to R,G,B**

Show the images as R,G,B components

**Note:**

-Experimental Procedure

RGB

(500x362)

**Color space conversion**

Y,Cg,Co

(500x362)

Inverse

color space conversion

Original

color image

**Decompose**

**Compose**



(500x362)

Reconstructed color image

-Flowers.bmp image

362

500

R,G,B

In this paper a new color space called Y,Cg,Co is suggested as part of the H.264 AVC (advanced video coding) standard.

- A new color space is defined by using the following basic equations:

Y=; Cg=; Co=

or not to introduce conversion rounding error by using the following equations:

Co=R-B; t=B+(Co>>1); Cg=G-t; Y=t+(Cg>>1) for the forward transformation

and t=Y-(Cg>>1); G=t+Cg; B=t-(Co>>1); R=B+Co for the inverse transformation

where t is an temporary variable and “>>” denotes an arithmetic right shift operation

**MATLAB SCRIPT:**

**Performing operation on girl256color.raw**

clc;

clear all;

close all;

img\_file = fopen('D:\STUDY\DIP\Test img\girl256color.raw','r');

Img\_read = fread(img\_file);

Red=Img\_read(1:3:length(Img\_read));

Green=Img\_read(2:3:length(Img\_read));

Blue=Img\_read(3:3:length(Img\_read));

%Converting row vector into a 2D matrix.

Red=(reshape(Red,256,256))';

Green=(reshape(Green,256,256))';

Blue=(reshape(Blue,256-,256))';

%Build Y I Q components

Y\_Lumi= Red\*0.299 + Green\*0.587 + Blue\*0.114;

I\_Chro= Red\*0.596 + Green\*-0.274 + Blue\*-0.322;

Q\_Chro= Red\*0.211 + Green\*-0.523 + Blue\*0.312;

figure(1);

subplot(2,3,1);

image(uint8(Y\_Lumi));

title('Y-Luminance ');

subplot(2,3,2);

image(uint8(I\_Chro));

title('I-Chrominance');

subplot(2,3,3);

image(uint8(Q\_Chro));

title('Q-Chrominance');

% Reconstruction of RGB components from Y I Q

recon\_R=Y\_Lumi+0.956\*I\_Chro+0.621\*Q\_Chro;

recon\_G=Y\_Lumi-0.272\*I\_Chro-0.647\*Q\_Chro;

recon\_B=Y\_Lumi-1.106\*I\_Chro+1.703\*Q\_Chro;

subplot(2,3,4);

image(uint8(recon\_R));

title('R-Image');

subplot(2,3,5);

image(uint8(recon\_G));

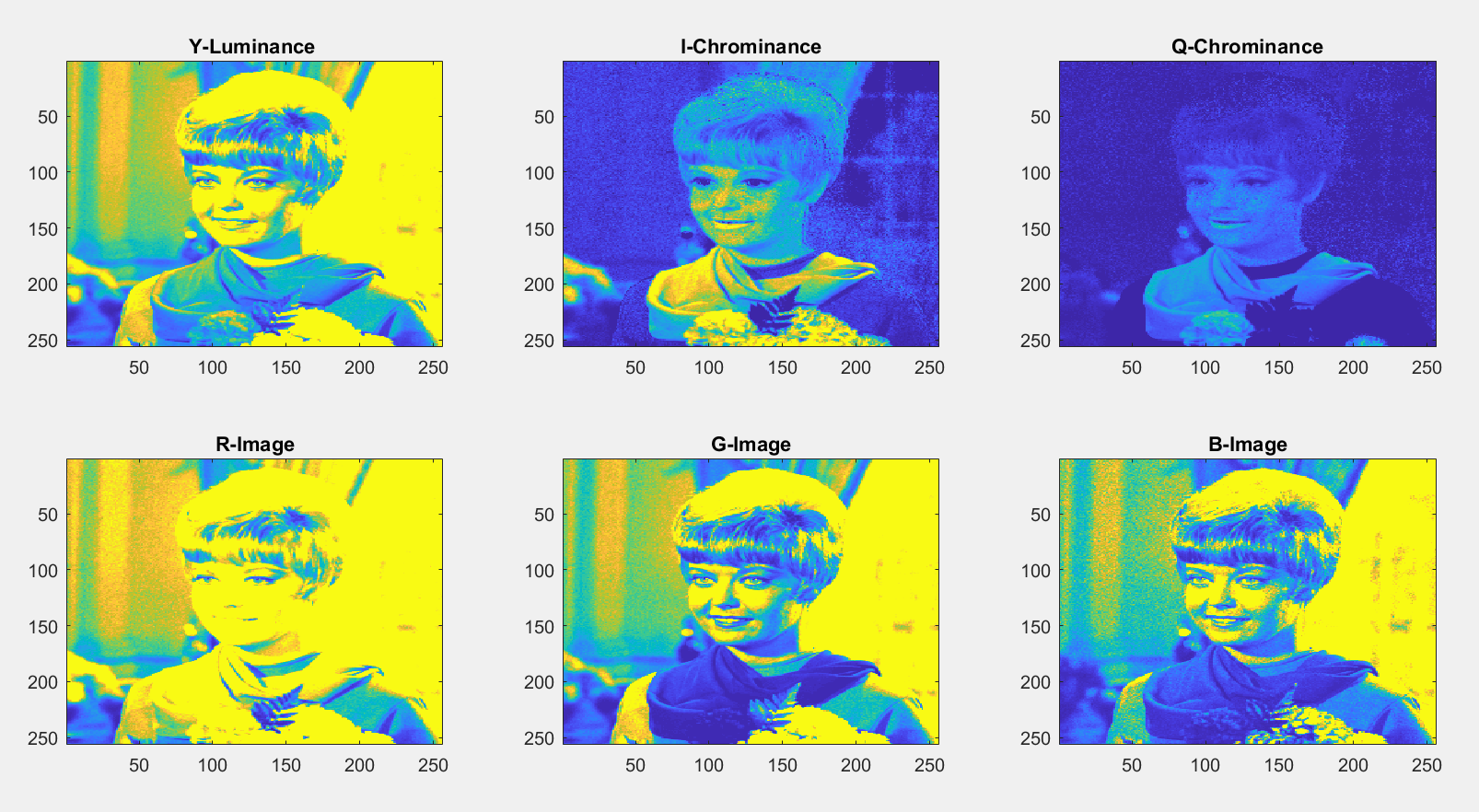
title('G-Image');

subplot(2,3,6);

image(uint8(recon\_B));

title('B-Image');

**OUTPUT:**



**MATLAB SCRIPT:**

**Performing operation on flowers.bmp image**

% input image = flowers.bmp

Img= double(imread('D:\STUDY\DIP\Test img\flowers.bmp'));

figure(2);

imshow('D:\STUDY\DIP\Test img\flowers.bmp');

title('original image')

%Separating RGB components.

Red=Img(:,:,1);

Green=Img(:,:,2);

Blue=Img(:,:,3);

%converting RGB into Y Cg Co

Y\_Lumi=1/2\*(Green + (Red+Blue)/2);

Cg\_Chro=1/2\*(Green - (Red+Blue)/2);

Co\_Chro=(Red- Blue)/2;

figure(3);

subplot(2,3,1);

image(uint8(Y\_Lumi));

title('Y - Image');

subplot(2,3,2);

image(uint8(Cg\_Chro));

title('Cg - Image');

subplot(2,3,3);

image(uint8(Co\_Chro));

title('Co - Image');

%RGB Components

R\_comp=2\*Y\_Lumi - 2\*Cg\_Chro + 2\*Co\_Chro;

G\_comp=Y\_Lumi + Cg\_Chro;

B\_comp=Y\_Lumi-Cg\_Chro-Co\_Chro;

subplot(2,3,4);

image(uint8(R\_comp));

title('R - Image');

subplot(2,3,5);

image(uint8(G\_comp));

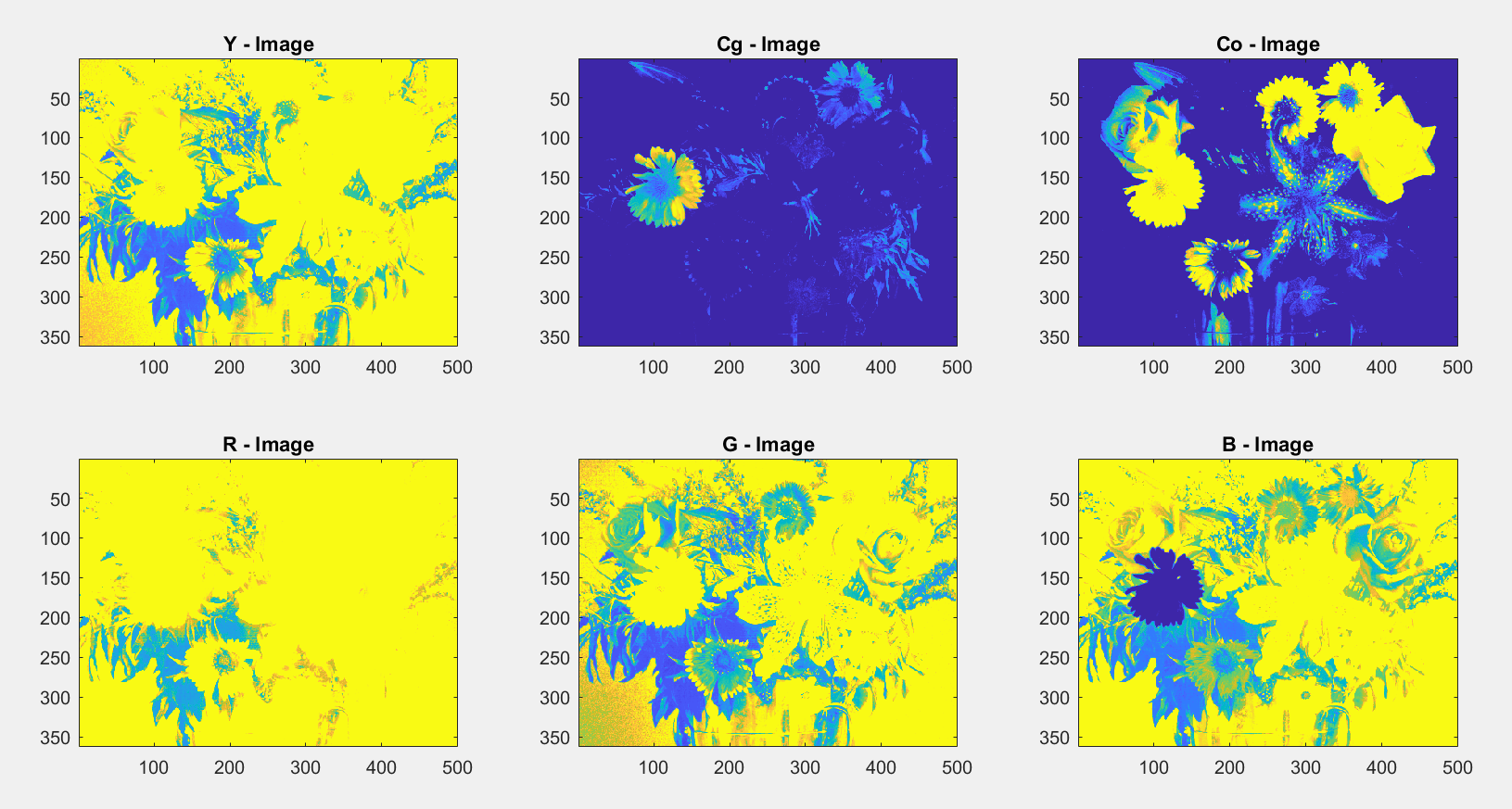
title('G - Image');

subplot(2,3,6);

image(uint8(B\_comp));

title('B - Image');

**OUTPUT: -**



**PROCEDURE:** -

* Initially we will open the image in matlab using **fopen** command the the input image is read using the **fread** command.
* The original input image is then displayed using the **imshow** command.
* Then we will separate the Red, Green and Blue color components of the original image.
* **MATLAB CODE FOR EXTRACTING THIS COMPONENTS: -**Red=Img\_read(1:3:length(Img\_read));

Green=Img\_read(2:3:length(Img\_read));

Blue=Img\_read(3:3:length(Img\_read));

* After extracting the desired color components we separate the Luminance(Y) , Hue(I) and Saturation(Q) components of the image.
* **MATLAB CODE FOR EXTRACTING THIS COMPONENT: -**

Y\_Lumi= Red\*0.299 + Green\*0.587 + Blue\*0.114;

I\_Chro= Red\*0.596 + Green\*-0.274 + Blue\*-0.322;

Q\_Chro= Red\*0.211 + Green\*-0.523 + Blue\*0.312;

* Thereafter we will reconstruct Red, Green and Blue components from Y, I and Q
* **MATLAB CODE FOR IMAGE RECONSTRUCTION: -**recon\_R=Y\_Lumi+0.956\*I\_Chro+0.621\*Q\_Chro;

recon\_G=Y\_Lumi-0.272\*I\_Chro-0.647\*Q\_Chro;

recon\_B=Y\_Lumi-1.106\*I\_Chro+1.703\*Q\_Chro;

* Then we will take the flower image and all the above steps will be repeated for the flower image.

**CONCLUSION:** -  
  
 In this project the main objective was to compare the original image with the reconstructed image that is obtained from Y, I and Q components. Thus, from the above results it can be inferred that there is some difference between the original and reconstructed image. This is because there is some error present in the reconstructed image that is obtained from the Y, I and Q components.