CE-712: Digital Image Processing of Remotely Sensed Data

Tutorial Exercise 3

Converting Image Storage Classes

For conversion of image storage classes, matlab has toolbox functions such as:

Filename = B1_mumbai

>>im2double(B1_mumbai);

>>im2uint8(B1_mumbai);

>>im2uint16(B1_mumbai);

Image Arithmetics

To add two images or add a constant value to an image use *imadd* function.

>> B1 = imread('B1_mumbai.tif');

>> B2 = imread('B2_mumbai.tif');

>> B3 = imadd(B1,B2);

To subtract one image from another use *imsubtract* function.

>> B1 = imread('B1_mumbai.tif');

>> B2 = imread('B2_mumbai.tif');

>> B4 = imsubtract(B1,B2);

To perform element to element multiplication of each corresponding pixel in a pair of input images use *immultiply*.

>> B1 = imread('B1_mumbai.tif');

>> B2 = imread('B2_mumbai.tif');

>> B5 = immultiply(B1,B2);

To perform element to element division of each corresponding pixel in a pair of input images use *immultiply*.

>> B1 = imread('B1_mumbai.tif');

>> B2 = imread('B2_mumbai.tif');

>> B6 = imdivide(B1,B2);

Colors in Matlab

MATLAB can display the values of any matrix as an image depending on the commands you use.

Displaying your matrix as an image in a variety of ways gives you further insight into your data. By understanding the different image types explained in this article, you'll know exactly how MATLAB turns matrix values into image pixel colors. You can then control the way MATLAB displays your data, and correct images that display incorrectly.

Image Types

An *image type* is a particular method of associating matrix values with pixel colors. MATLAB has three basic image types:

- **Truecolor**--generated by digital cameras and widely used in computer graphics
- **Indexed** and **scaled indexed**--often used to display scientific and engineering data with an associated color scale representing the data units

The Image Processing Toolbox identifies two additional types:

- **Grayscale**--often used in image processing and image analysis algorithms
- **Binary**--often used as a mask to indicate segmentation results or a region of interest

Truecolor Images

In a truecolor image, every image pixel has three values associated with it: the red, green, and blue components. You represent a truecolor image in MATLAB with a three-dimensional array of size M-by-N-by-3. Display functions in MATLAB and the Image Processing Toolbox treat such an array as a truecolor image.

For example, let's construct a truecolor image with three rows and three columns.

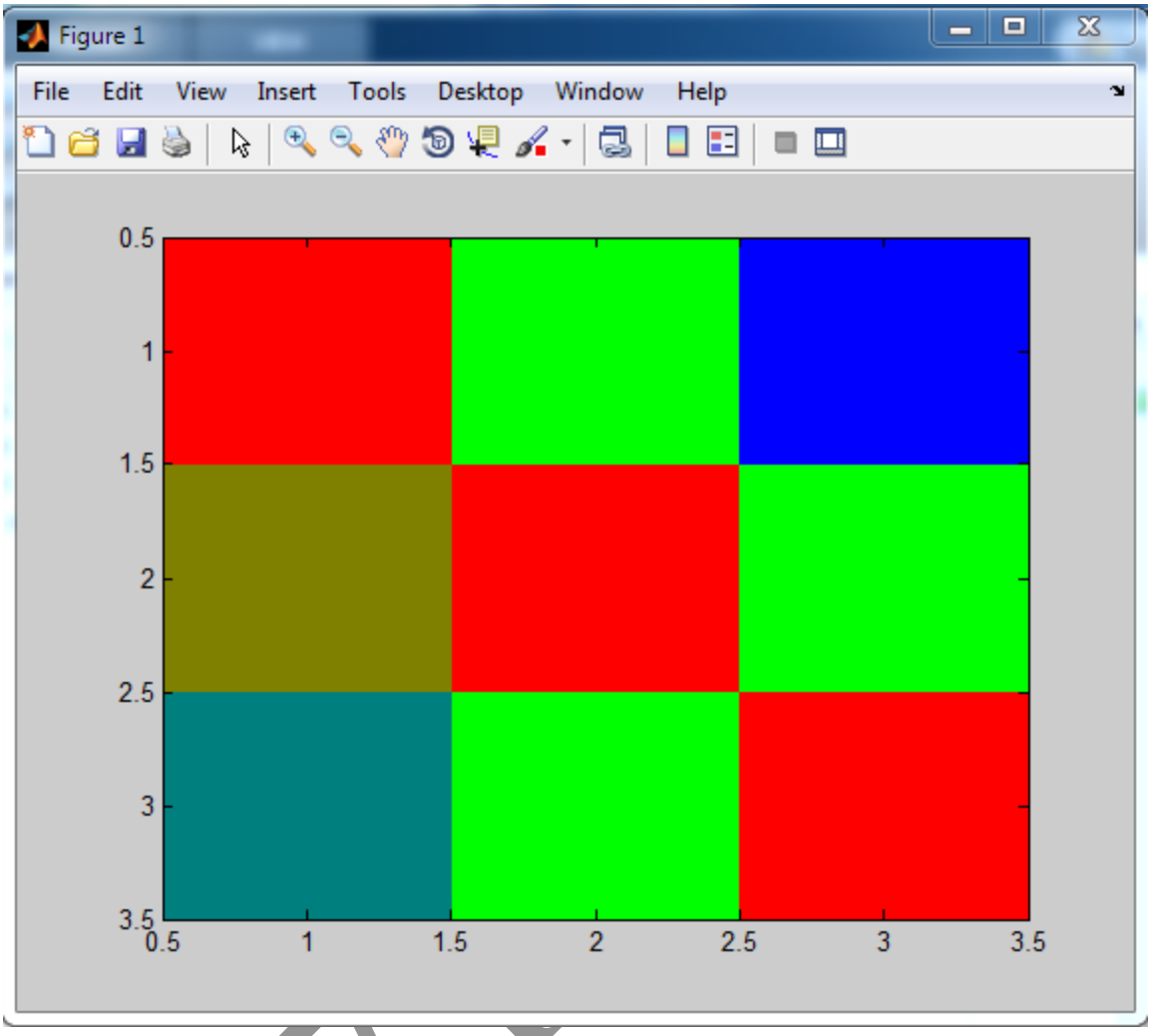
```
>> r = [1 0 0; 0.5 1 0; 0 0 1];
>>g = [0 1 0; 0.5  0 1; 0.5 1 0];
>>b = [0 0 1; 0 0 0; 0.5 0 0];
```

Concatenate the above arrays to form a 3D rgb image.

```
>> rgb = cat (3,r,g,b);
```

Display the Above true color image using imagesc command

```
>> figure;
>> imagesc(rgb);
```



It is evident from the output that the first pixel represents the red color combination [1,0,0] second pixel represents a green color combination [0,1,0] and the third pixel represents a blue color combination [0,0,1] other pixel(2,1) is combination of rgb.

To generate Custom colormap in matlab

Matlab represents colors in m x 3 matrix where m are the number of rows and 3 is the rgb combination

To create a custom random colormap create a 256x3 random matrix as follows.

>> colormap_random = rand(256,3);

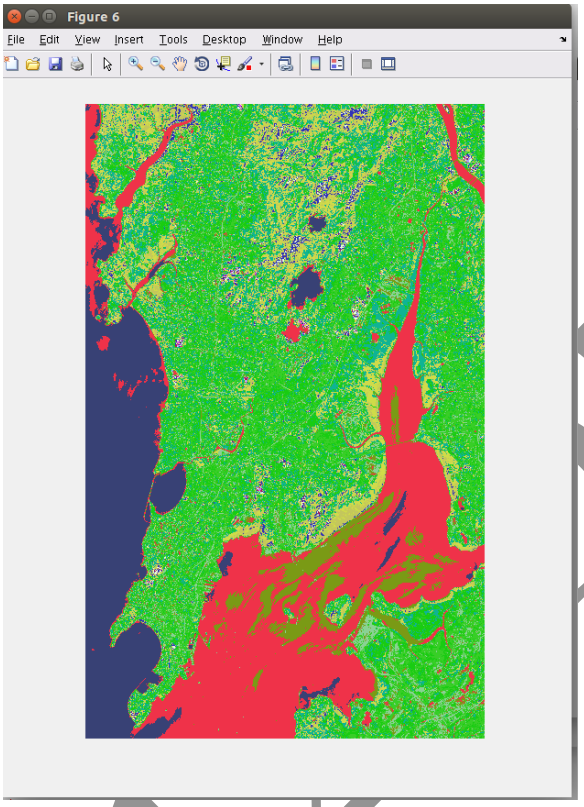
Use this colormap to represent gray scale single band data as pseudo color image

>> figure;

>>imshow(B5_mumbai);

>>colormap(colormap_random);

The figure below displays Pseudo color for band 6 of Landsat 8



To create a custom rgb colormap use the following code

T = [255 , 0 , 0

0 , 255 , 0

255, 0 , 0]./255

X = [0

128

255]

Colormap_rgb = interp1(x/255,T,linspace(0,1,256));

To display color composites in Matlab

Stack bands 4,3,2 from landsat 8 for Mumbai region using cat function

>> rgb = cat(3,B4_mumbai,B3_mumbai,B2_mumbai);

To display the true color composite use:

>> imshow(rgb);

To display False color composites (FCC) stack the bands of interest, for e.g for bands 5,4,3

>> b543 = cat(3,B5_mumbai,B4_mumbai,B3_mumbai);

>> imshow(b543);

To scale the image use the immultiply function :

>>immultiply(rgb,2);

Display the image and check the difference in the images.

For Loop to display multiple images in matlab

To process and display multiple images in matlab we can use for loop as discussed below.

First import all bands for Mumbai region to matlab.

Stack the complete band using the *cat()* function to obtain a single file containing all band data(**don't stack band 8 image**)

>> all_band = cat(3,B1,B2,B3,B4,B5,B6,B7,B9,B10,B11);

Syntex for FOR LOOP

for i = 1:10;

band = all_band(:,:,i);

figure;

imshow(band);

colormap(colormap_random);

end

NOTE: To execution the above FOR loop use the Matlab Editor to write a script.

Assignment

1) Create a 3x3x3 matrix for 8 bits containing red (255,0,0) , green(0,255,0), blue(0,0,255) and for 16 bits and display the true color composite for both the arrays.
2) What is the purpose of the interp1 function?
3) Can we change the colormap for true color? If no, why? If yes, how?
4) Create a True color composite for the Mumbai region using Landsat 8 image and multiple the matrix by 2 , 4 and display them. What is the difference in the images? Check the histograms of three images and write the conclusions.
5) Create a FCC for the Mumbai region and repeat the steps in Question 3.
6) Create a FOR loop to Display Pseudo random color for each individual bands of Landsat 8 (Mumbai region).

Note:- Please take the screen shots of all the matrix operation commands and results and attach it with your submission.