**Report Type: Analytical assignment report**

**Name and Surname: Aya Ahmed Faeek ELGebaly**

**Student Number: 19017602**

**RESEARCH:**

In this problem we have to take the artificially colored MRI image of a brain, which contains a red tumor, as an input and separate it from the image. With the help of the Image segmentation technique [1] we can separate the tumor. Image segmentation technique is a commonly used technique to partition an image into multiple parts or regions. It can involve separating foreground from background, or clustering regions of pixels based on similarities in color or shape. When we come to our code, first we have to read our input image by imread()[3] function. Our image is RGB (True color) Image [2], which is stored as an m-by-n-by-3 data array that defines red, green, and blue color components for each individual pixel as shown in Figure 1.

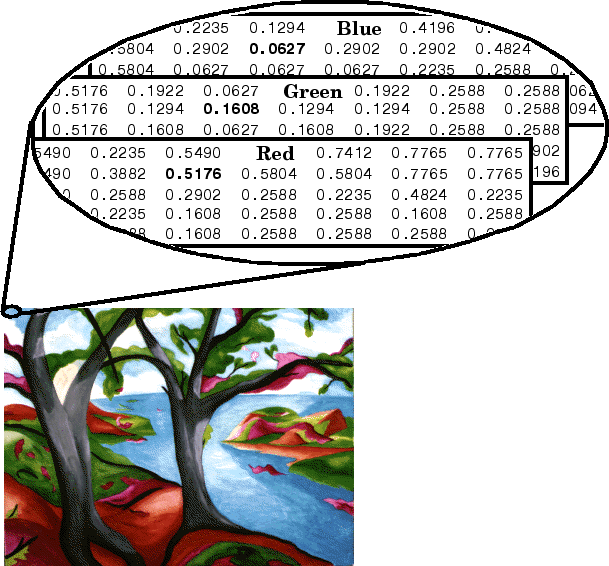


Figure 1 – RGB (True color) Image

After reading our image I separated each date array of the RGB image. Then I multiplied them by ‘zero’ to get rid of its data. And after that I can assign only the wanted data with ‘one’. The pixel whose color components are (0,0,0) is displayed as black, and a pixel whose color components are (1,1,1) is displayed as white. We can identify the threshold values(wanted date) by optimizing them from our original image by trial and error. I found the red value is greater than or equal to 110, the green value is smaller than or equal 49 and the blue value is smaller than or equal 31.

To create a segmentation mask I have to create a 3\_D matrix, like our original image as it is an RGB Image, to agree with the matrix multiplication process.

* First, we have to multiply our three layers (red, green, blue) to find the intersection between them.
* Second, when we come to the third dimension we have to put the wanted data in our RGB layers.

Now we have a segmentation mask consisting of only ones and zeros, and now we can multiply it by the original image to get the output image or segmented image. In the subplot part; I multiplied the segmentation mask by 1000 to make it visible.

**CODE:**

| %% start  close all  clear all  clc    %% Load the image    input\_image = imread('mri\_img.png');      %% obtain the three color channels and assign them to zero to get rid of all of the elements    Red\_slide = 0\*input\_image(:,:,1);  Green\_slide = 0\*input\_image(:,:,2);  Blue\_slide = 0\*input\_image(:,:,3);      %% Then use find function to assign the wanted elements  r = find(input\_image(:,:,1)>=110);  g = find(input\_image(:,:,2)<=49);  b = find(input\_image(:,:,3)<=31);    Red\_slide(r) = 1;  Green\_slide(g) = 1;  Blue\_slide(b) = 1;    %% Use element-wise multiplication to make a composition of the three layers    segmentation\_mask = Red\_slide.\*Green\_slide.\*Blue\_slide; %to take the intersection of matrix  segmentation\_mask\_3d(:,:,1) = segmentation\_mask;  segmentation\_mask\_3d(:,:,2) = segmentation\_mask;  segmentation\_mask\_3d(:,:,3) = segmentation\_mask; %the segmetation mask consist of only ones and zeroes    %% the segmentation of the image  output\_image = segmentation\_mask\_3d.\*input\_image;    %% use subplot to show Original Image, Segmentation Mask and Segmented Image side by side  figure,subplot(1,3,1),imshow(input\_image),title('Original Image','fontsize',12)  subplot(1,3,2),imshow(1000\*segmentation\_mask),title('Segmentation Mask','fontsize',12)  subplot(1,3,3),imshow(output\_image),title('Segmented Image','fontsize',12) |
| --- |

**COMMENTS ABOUT THE CODE:**

The code explains what is done. And I tried to explain why I do each step by writing comments inside the code using the comment sign ‘%’.As I started my code by loading the image by using the imread function. After that I worked to obtain the three color layers of the RGB image and assign them to zero to get rid of all of the data on them. And using find function and threshold values I could assign the wanted date in our “red\_green\_blue” layers again. But this time we have only the wanted date. Then, I used element-wise multiplication to make a composition of the three layers to take the intersection of the matrix. Now we have our segmentation mask which consists of only ones and zeroes. And it is ready to convert our original image to a segmented image. Finally, by using subplot we can show Original Image, Segmentation Mask and Segmented Image side by side

**RESULTS:**

In the result part we can see our original (input image) and the segmentation mask which multiplied by it to produce the segmented image (output image) as shown in Figure 2.

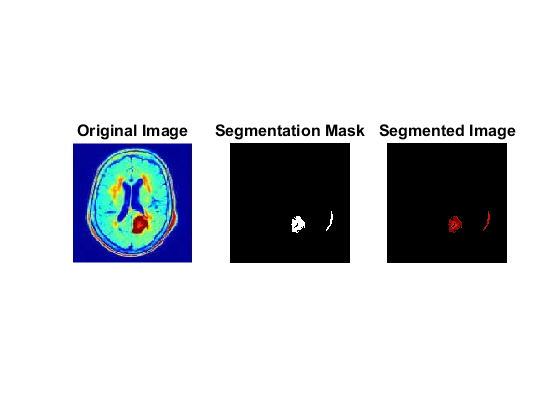
****

Figure 2 – the original, segmentation mask and segmented image.

**COMMENTS ABOUT THE RESULTS:**

It is a clear segmented image which means that our step is right and our threshold values are accurate. And in the subplot part; I multiplied the segmentation mask by 1000 to make it visible. And now we can see it clearly.