

Content Based Image Recognition – CSE3018

Lab 1 – 30/11/2017

Name: Ayush Sharma

Reg. No: 15BCE1335

Faculty: Dr. S Geetha

1. Read a color image and find its dimension
2. Read a grayscale image and find its dimension (or convert color to grayscale)
3. Separate into color planes (RGB)
4. Display all images in a single plot
5. Convert image into black and white
6. Bit Plane separation
7. Find an image which resembles from 0th bit.
8. Edge Detection Techniques

Code and Screenshots:

```
%Task 1 & 2
%Reading an image file and finding its dimensions
A = 'C:\Users\TEMP.VITUCC\Desktop\Tulips.jpg';
B = imread(A, 'jpg');
figure(1), imshow(B), title('Original Image');
```



```
C = rgb2gray(B);
figure(2), imshow(C), title('Grayscaled Image');
```



```
imwrite(C, 'C:\Users\TEMP.VITUCC\Desktop\tulips_rbg.jpg','jpg');
B_size = size(B) %size of color image
C_size = size(C) %size of grayscale image
```

```
%Task 3
%Separating into 3 color planes
IMG_RED = B(:,:,1);
IMG_GREEN = B(:,:,2);
IMG_BLUE = B(:,:,3);
figure(3), imshow(IMG_RED), title('Red Plane');
figure(4), imshow(IMG_GREEN), title('Green Plane');
figure(5), imshow(IMG_BLUE), title('Blue Plane');
```



Green Plane



Blue Plane



```
%Task 4
%Display all images in a single plot
figure (6);
subplot(2,3,1); imshow(B), title('Original Image');
subplot(2,3,4); imshow(IMG_RED), title('Red Plane');
subplot(2,3,5); imshow(IMG_GREEN), title('Green Plane');
subplot(2,3,6); imshow(IMG_BLUE), title('Blue PLane');
```

Original Image



Red Plane



Green Plane



Blue PLane



```
%Task 5
%Convert image into Black and White
IMG_BW = im2bw(B);
BW_size = size(IMG_BW)
figure(7), imshow(IMG_BW), title('Black and White');
```

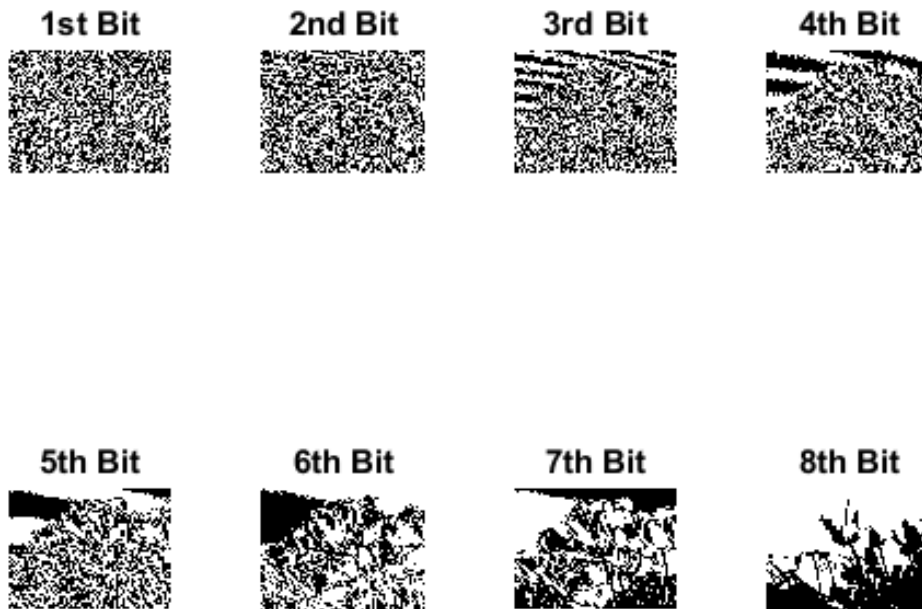
Black and White



```

%Task 6
%Bit plane Separation
IMG_BIT1 = double(bitget(C,1));
IMG_BIT2 = double(bitget(C,2));
IMG_BIT3 = double(bitget(C,3));
IMG_BIT4 = double(bitget(C,4));
IMG_BIT5 = double(bitget(C,5));
IMG_BIT6 = double(bitget(C,6));
IMG_BIT7 = double(bitget(C,7));
IMG_BIT8 = double(bitget(C,8));
figure(8);
subplot(2,4,1); imshow(IMG_BIT1), title('1st Bit');
subplot(2,4,2); imshow(IMG_BIT2), title('2nd Bit');
subplot(2,4,3); imshow(IMG_BIT3), title('3rd Bit');
subplot(2,4,4); imshow(IMG_BIT4), title('4th Bit');
subplot(2,4,5); imshow(IMG_BIT5), title('5th Bit');
subplot(2,4,6); imshow(IMG_BIT6), title('6th Bit');
subplot(2,4,7); imshow(IMG_BIT7), title('7th Bit');
subplot(2,4,8); imshow(IMG_BIT8), title('8th Bit');

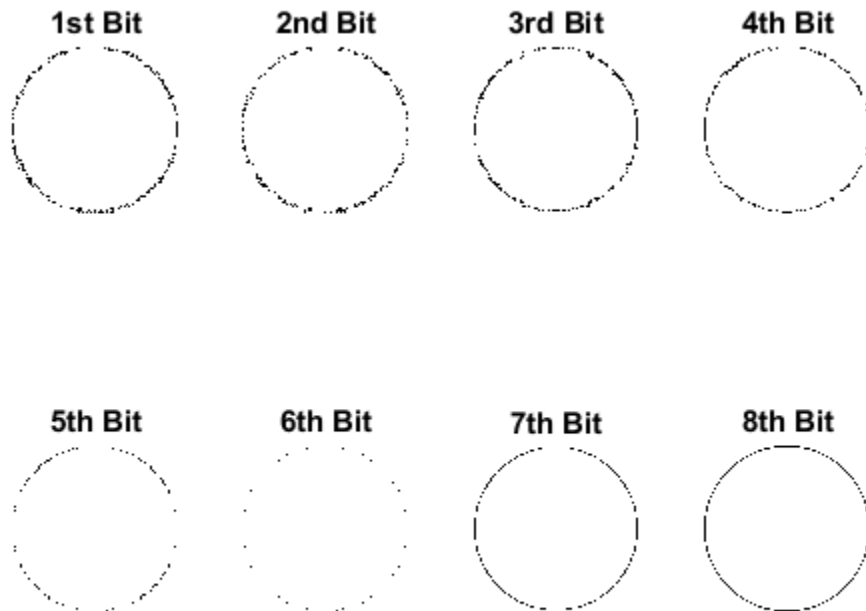
```



```

%Task 7
%Find and image with significant 0th bit
D = 'C:\Users\TEMP.VITUCC\Desktop\Circle.jpg';
E = imread(D, 'jpg');
IMG1_BIT1 = double(bitget(E,1));
IMG1_BIT2 = double(bitget(E,2));
IMG1_BIT3 = double(bitget(E,3));
IMG1_BIT4 = double(bitget(E,4));
IMG1_BIT5 = double(bitget(E,5));
IMG1_BIT6 = double(bitget(E,6));
IMG1_BIT7 = double(bitget(E,7));
IMG1_BIT8 = double(bitget(E,8));
figure(9);
subplot(2,4,1); imshow(IMG1_BIT1), title('1st Bit');
subplot(2,4,2); imshow(IMG1_BIT2), title('2nd Bit');
subplot(2,4,3); imshow(IMG1_BIT3), title('3rd Bit');
subplot(2,4,4); imshow(IMG1_BIT4), title('4th Bit');
subplot(2,4,5); imshow(IMG1_BIT5), title('5th Bit');
subplot(2,4,6); imshow(IMG1_BIT6), title('6th Bit');
subplot(2,4,7); imshow(IMG1_BIT7), title('7th Bit');
subplot(2,4,8); imshow(IMG1_BIT8), title('8th Bit');

```



```

%Task 8
%Apply Edge detection techniques on 4 different types of images
%Techniques: Sobel, Prewitt, Roberts, Canny.
%Images: Building, Car, Human Crowd, Nature
%
Building = 'C:\Users\TEMP.VITUCC\Desktop\building.jpg';
Car = 'C:\Users\TEMP.VITUCC\Desktop\car.jpg';
Human_Crowd= 'C:\Users\TEMP.VITUCC\Desktop\human_crowd.jpg';
Nature = 'C:\Users\TEMP.VITUCC\Desktop\nature.jpg';
IMG_Building = imread(Building, 'jpg');
IMG_Car = imread(Car, 'jpg');
IMG_Crowd = imread(Human_Crowd, 'jpg');
IMG_Nature = imread(Nature, 'jpg');

IMG_Building = rgb2gray(IMG_Building);
BUILD_SOBEL = edge(IMG_Building, 'Sobel');
BUILD_PREWITT = edge(IMG_Building, 'Prewitt');
BUILD_ROBERTS = edge(IMG_Building, 'Roberts');
BUILD_CANNY = edge(IMG_Building, 'Canny');
figure(10);
subplot(3,2,1); imshow(IMG_Building), title('Building');
subplot(3,2,3); imshow(BUILD_SOBEL), title('Sobel');
subplot(3,2,4); imshow(BUILD_PREWITT), title('Prewitt');
subplot(3,2,5); imshow(BUILD_ROBERTS), title('Roberts');
subplot(3,2,6); imshow(BUILD_CANNY), title('Canny');

```


Building



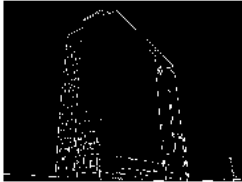
Sobel



Prewitt



Roberts



Canny



```
IMG_Car = rgb2gray(IMG_Car);
BUILD_SOBEL = edge(IMG_Car, 'Sobel');
BUILD_PREWITT = edge(IMG_Car, 'Prewitt');
BUILD_ROBERTS = edge(IMG_Car, 'Roberts');
BUILD_CANNY = edge(IMG_Car, 'Canny');
figure(11);
subplot(3,2,1); imshow(IMG_Car), title('Car');
subplot(3,2,3); imshow(BUILD_SOBEL), title('Sobel');
subplot(3,2,4); imshow(BUILD_PREWITT), title('Prewitt');
subplot(3,2,5); imshow(BUILD_ROBERTS), title('Roberts');
subplot(3,2,6); imshow(BUILD_CANNY), title('Canny');
```

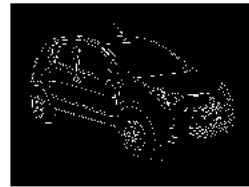
Car



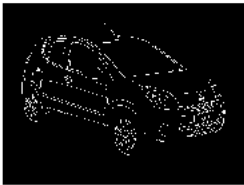
Sobel



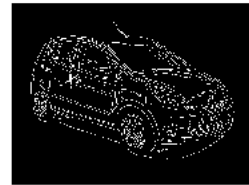
Prewitt



Roberts



Canny

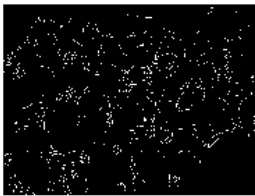


```
IMG_Crowd = rgb2gray(IMG_Crowd);
BUILD_SOBEL = edge(IMG_Crowd, 'Sobel');
BUILD_PREWITT = edge(IMG_Crowd, 'Prewitt');
BUILD_ROBERTS = edge(IMG_Crowd, 'Roberts');
BUILD_CANNY = edge(IMG_Crowd, 'Canny');
figure(12);
subplot(3,2,1); imshow(IMG_Crowd), title('Crowd');
subplot(3,2,3); imshow(BUILD_SOBEL), title('Sobel');
subplot(3,2,4); imshow(BUILD_PREWITT), title('Prewitt');
subplot(3,2,5); imshow(BUILD_ROBERTS), title('Roberts');
subplot(3,2,6); imshow(BUILD_CANNY), title('Canny');
```

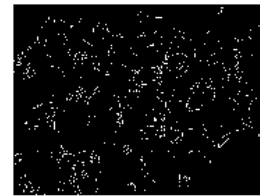
Crowd



Sobel



Prewitt



Roberts



Canny



```
IMG_Nature = rgb2gray(IMG_Nature);
BUILD_SOBEL = edge(IMG_Nature, 'Sobel');
BUILD_PREWITT = edge(IMG_Nature, 'Prewitt');
BUILD_ROBERTS = edge(IMG_Nature, 'Roberts');
BUILD_CANNY = edge(IMG_Nature, 'Canny');
figure(13);
subplot(3,2,1); imshow(IMG_Nature), title('Nature');
subplot(3,2,3); imshow(BUILD_SOBEL), title('Sobel');
subplot(3,2,4); imshow(BUILD_PREWITT), title('Prewitt');
subplot(3,2,5); imshow(BUILD_ROBERTS), title('Roberts');
subplot(3,2,6); imshow(BUILD_CANNY), title('Canny');
```

Nature



Sobel



Prewitt



Roberts



Canny

