**Image Processing Assignment**

1. Define Image. How it can be represent in 2 dimensions?
2. Describe image model?
3. What do you mean by IP?
4. Explain various methods for Image processing?
5. Elaborate image processing environment?
6. Explain the various fields of IP?
7. How will you represent DIP?
8. Explain Resolution? Write the factors also
9. Explain DIP operations?
10. Explain the elements of DIP
11. Explain the stets involved in DIP
12. Load the image ‘cameraman.tif’, find its dimensions and number of channels and display it.
13. Load the image ‘cameraman.tif’, and save it in ‘jpeg’ format with a compression factor of 25. Display the two images (original and compressed) and compare them visually. What do you observe?

Using the function ‘imfinfo’, find the size of files before and after compression and hence compute the compression ratio.

1. What happens if you convert a double having values out side the range [0 255] to an uint8?
2. Load the image ‘rice.png’, binarize it using the function ‘im2bw’ with a threshold of 0.5 and display both original and binarized images.
3. Write a small program in Matlab to hide the gray image in the colored image as already discussed.
4. Once encoded, add statements to your program to extract the gray image from the colored image.
5. Display the original, encoded and extracted images in your program.
6. Obtain the images “lena.bin” and “peppers.bin” from the course web site. Each image has 256 × 256 pixels and each pixel has 8 bits.

(a) Read and display the images.

(b) Define a new 256 × 256 image J as follows: the left half of J, i.e., the first 128 columns, should be equal to the left half of the Lena image. The right half of J, i.e., the 129th column through the 256th column, should be equal to the right half of the Peppers image.

(c) Define a new 256 × 256 image K by swapping the left and right halves of J.

(d) Be sure to turn in: A listing of your code and printouts of the original images, image J, and image K.

1. Use Matlab for this problem.

(a) Type help imread and help imwrite at the Matlab prompt to read the online help for these commands.

(b) Obtain the image “lenagray.jpg” from the course web site. It is the same image that you used in the first problem, but the file is in JPEG format this time.

 (c) Use the imread function to read in the image. Let’s call this image J1.

 (d) Make a new image J2 that is the photographic negative of J1. To do this, set J2 = 255−J1. Display the new image J2 and use the imwrite command to write it out as a JPEG file.

 (e) Be sure to turn in: A listing of your code and printouts of the original and modified images.

1. Use Matlab for this problem.

(a) Obtain the color image “lena512color.jpg” from the course web site. It is the same image that you used in the first two problems, except this time it is in color (each pixel has 24 bits) and the size is 512 × 512 pixels. If you read the image into a Matlab array J1, then J1(:,:,1) is the Red band, J1(:,:,2) is the Green band, and J1(:,:,3) is the Blue band. In each band, each pixel has 8 bits, just like the image in the first problem.

(b) Use imread to read in the image and then display it. Let’s call this image J1.

(c) Make a new color image J2 by swapping the color bands of J1 as follows. First, just set J2 = J1 to initialize the new image with the right size. Then make the Red band of J2 equal to the Blue band of J1, make the Green band of J2 equal to the Red band of J1, and make the Blue band of J2 equal to the Green band of J1. For example, to set the Red band of J2 equal to the Blue band of J1, you can type J2(:,:,1) = J1(:,:,3);.

(d) Display the new image and use imwrite to write it out to a JPEG file.

(e) Be sure to turn in: A listing of your code and printouts of the original and modified images.

1. Obtain the image “Mammogram.bin” from the course web site. This image has 256 × 256 pixels. Each pixel has 8 bits. Note: the server is Unix; the filename is case sensitive. Do not make the mistake of getting the incorrect 512 × 512 file “mammogram.bin.”

 (a) There are two main regions in the input image: the imaged tissue and the dark background region on the left side of the image. Write a program to convert this gray scale image into a binary image by simple thresholding. In the binary image, use a value of 255 = 0xff for logical one and a value of 0 = 0x00 for logical zero. Select the threshold so that the binary image is equal to logical zero over the background region and logical one over the tissue.

(b) Write a program to implement the Approximate Contour Image Generation algorithm given on page 2.104 of the notes. Your program should input the binary image and output a binary contour image. Run your program to generate an appoximate contour image from the binary image you obtained by thresholding Mammogram.bin.

(c) Could a chain code be used to represent the main contour in your contour image? Why or why not? 2. Obtain the image “lady.256” from the course web site. This is a 256 × 256 gray scale image with 8-bit pixels. Plot a histogram for the image. Write a program to perform a full-scale contrast stretch on the image and plot a histogram for the result.

1. Obtain the image “actontBin.bin” from the course web site. This image has 256 × 256 pixels with 8 bits each. It is a true binary image; the pixel value 255 represents logical one and the pixel value 0 represents logical zero. Write a program to find instances of the letter “T” in the image using the Binary Template Matching algorithm given on pages 2.92 - 2.97 of the notes. You will have to design the template yourself based on an analysis of the image. Apply the match measure M2 at every pixel in the input image where a sufficiently large neighborhood exists. Construct an output image J1 where each pixel is equal to the match measure M2 (set J1 equal to zero at pixels where a sufficiently large neighborhood does not exist in the input image). Threshold the image J1 to obtain a binary image J2 that should be equal to logical one at pixels where there is a high probability that the letter “T” is present in the input image.
2. Obtain the image “johnny.bin” from the course web site. This image has 256 × 256 pixels. Each pixel has 8 bits. Plot the histogram of the original image. Write a program to perform histogram equalization on this image. Show the equalized image and plot its histogram.