LAB LESSON THREE

IMAGE ENHANCEMENT USING POINT PROCESSING OPERATIONS

Dr. Nursuriati Jamil

Image Enhancement

The purpose of image enhancement is to improve the appearance of an image so that it looks subjectively better. Removing blurring noise, increasing contrast and revealing details are examples of enhancement operations (Rapp, 1996). One of the most basic ways to enhance an image is to perform point processing operations on the image such as histogram processing and applying transformation functions.

Histogram Processing

The easiest way to enhance an image is to change its brightness and contrast. You can change these attributes by stretching the color distribution, equalizing the distribution of colors to utilize the full range of colors and adjusting the scaling of the colors.

Histogram of an image is a discrete function that is formed by counting the number of pixels in the image that has a certain gray values (Petrou & Bosdogianni, 2003). The gray values (brightness) are represented by the x-axis and the frequency of pixel occurrences are displayed as y-axis.

To view the current level of contrast in an image, you can use the histogram function of the Image Processing Toolbox, `imhist`. This function shows the distribution of gray levels in the specified image

```
I = imread('pout.tif');
imshow(I)
figure, imhist(I)
```

Histogram Stretching

Histogram stretching involves identifying minimum and maximum brightness values from the histogram and applying a transformation to uniformly stretch this range to fill the full range. This enhances the contrast in the image with light toned areas appearing lighter and dark areas appearing darker, differentiating objects and the background more effectively.

One way to increase the contrast of an image is to stretch the pixel values to the full range. Within a `unit8` image the full range would be from zero to 255.

$$J = 255\left(\frac{I - I_{\min}}{I_{\max} - I_{\min}}\right)$$

```
I = imread('pout.tif');
subplot(2,2,1), imshow(I); title('Original')
subplot(2,2,2), imhist(I);
Id = double(I);
I_min = min(Id(:));
I_max = max(Id(:));
J = 255*(Id-I_min)/(I_max-I_min);
J = uint8(J);
subplot(2,2,3), imshow(J); title('Stretched')
subplot(2,2,4), imhist(J);
```

Histogram Equalization

A uniform distribution of the range of values across the full range may not be the appropriate enhancement for certain application, especially if the input range is not uniformly distributed. An alternative is to use histogram equalization, which is similar to stretching. Equalization is a technique where the histogram of the resultant image is as flat as possible; with stretching the overall shape of the histogram remains the same (Umbaugh 1999).

```
J1 = histeq(I);
imhist(J);
figure, imshow(J)
```

Histogram Adjustment

Though histogram equalization increases the contrast of an image, it does not brighten an image. To brighten or darken an image, you can adjust the histogram to a new range. This allows the intensity of the image to be adjusted to a specified range. In the Image Processing Toolbox you can use the `imadjust` function to adjust the intensity of an image.

```
I = imread('cameraman.tif');
J = imadjust(I,[0 0.2],[0.5 1]);
imshow(I);
figure, imshow(J);
```

```
i=imread('Mammogram.jpg');
g1=imadjust(i,[0 1],[1 0]);
figure, imshow(g1);
g2=imadjust(i,[0.5 0.75],[1 0]);
figure, imshow(g2);
imhist(i);
g2=imadjust(i,[],[],2);
figure, imshow(g2);
```

In the *i*`madjust` command above, 0 refers to the low intensity in the input image I and 0.2 refers to the high intensity of the input image I. The values 0.5 and 1 refer to the low and high intensity of the output image J, respectively.

Graylevel Transformation Functions

Negative

The complement of a grayscale image is its photographic negative. Its negative image may be obtained by using `imcomplement` function.

```
i=imread('mammogram.jpg');
j=255-i; imshow(j);

negI = imcomplement(i);  figure, imshow(negI);
```

Log Transformations

Purpose of log transformation function is to spread or compress a narrow range of the low intensity values in the input image into a wider range of intensities.

```
i=imread('FourierSpectrum.jpg');
imshow(i);
g = log(1 + double(i));
figure, imshow(g,[]);
```

Gamma Correction

Gamma correction can be applied using imadjust function, a value of 0 – 1 given to the gamma parameter will brighten the image and a negative value will darken the image.

```
i=imread('Spine.jpg'); imshow(i);
j = imadjust(i, [ ], [ ], 0.6);
figure, imshow(j);
```

Bit Plane Slicing

Grayscale images can be transformed into a sequence of binary images by breaking them up into their bit-planes.

```
i = imread('cameraman.tif');
b0=mod(i,2);
imshow(b0, []);
b5 = mod(floor(i/32), 2);
imshow(b5, []);
```

Assignment

1. Perform bit-plane slicing on 'cameraman.tif' and display all the bit planes in one figure.
   (5 marks)
2. Extract or highlight the rock object from 'grayrock.jpg' image.
   (5 marks)
3. Enhance 'darkbean.jpg' as much as possible.
   (5 marks)

References:

Rapp, C.S. & Joyner, W.L. 1996. Image processing and image enhancement. http://www.theaps.org/education/k12curric/activities/1996-Labs/rapp.pdf [25 Dec 2003].

Petrou, M. & Bosdogianni, P. 2003. *Image processing: The fundamentals.* Chichester: John Wiley & Sons Inc.

Umbaugh, S. 1999. *Computer vision and image processing.* Los Angeles: Prentice Hall Inc.