**LAB 4 Algebra and Logical Operation**

**Objectives:**

1. To learn more basic functions of *MATLAB Image Processing Toolbox*.

2. To do exercises on algebra and logical operation

* **Basic functions of IMT**

***Functions to practice:***

1. **Imshow:**

**f = imread('coins.png');**

**imshow(f);**

**figure;**

**imshow(f, 'InitialMagnification', 150);**

**figure;**

**imshow(f, 'InitialMagnification', 50);**

**imshow('coins.png','Border','tight');**

**[X1,map1]=imread('lena.jpg');**

**[X2,map2]=imread('baboon.jpg');**

**subplot(1,2,1), imshow(X1,map1)**

**subplot(1,2,2), imshow(X2,map2)**

1. **Imtool:**
2. **imrotate:**

**X1 = imread('lena.jpg');**

**X2 = imrotate(X1, +/-30);**

**imshow(X1)**

**figure, imshow(X2)**

1. **imcrop:**

**X1 = imread('lena.jpg');**

**X2 = imcrop(X1, [60 40 100 90]);**

**Imshow(X2)**

* **Algebra and Logical Operation**

***Algebra Operation***

*C(x, y)=A(x, y) + B(x, y)*

*C(x, y)=A(x, y) - B(x, y)*

*C(x, y)=A(x, y) \* B(x, y)*

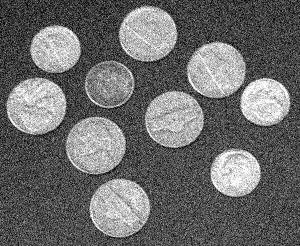
*C(x, y)=A(x, y)÷ B(x, y)*

1. **Addition**

* **Gaussian noise** is statistical noise, the values that the noise can take on are Gaussian-distributed.
* **white Gaussian noise**, in which the values at any pairs of times are statistically independent (and uncorrelated)
* **‘salt and pepper’** **noise**: on and off pixels

**imnoise(f,** '**gaussian**'**, 0.02);**

**imnoise(f,** '**salt & pepper**'**, 0.02);**

** **

**Gaussian Noise Salt and pepper noise**

Assume: *M* still images with noises

***Ci(x, y) = S(x, y) + Ni(x, y)***



1. **Subtraction**

**imsubtract(f1, f2);**

1. **Multiplication**

**Immultiply(f1, 1.2);**

Multiply 2, 3 to an image, respectively.

1. **Division**

**f1 = imread('rice.png');**

**f2 = imdivide(f1,2);**

**figure, imshow(I)**

**figure, imshow(J)**

***Logical Operation***

and, or, not

**and(imagefile1, imagefile2);**

* **Connected Components Labeling**

The bwlabel and the bwlabeln functions perform connected-component labeling, which is a method for identifying objects in a binary image. The bwlabel function supports 2-D inputs only; the bwlabeln function supports inputs of any dimension.

These functions return a matrix, called a label matrix. A label matrix is an image, the same size as the input image, in which the objects in the input image are distinguished by different integer values in the output matrix.

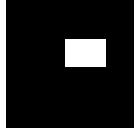
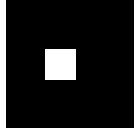
The label matrix returned by bwlabel or bwlabeln is of class double; it is not a binary image. One way to view it is to display it as a pseudocolor indexed image, using label2rgb.

**Practice: Generate the following image. Label the binary image. Request to show the original and labeled images.**

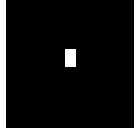
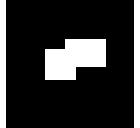
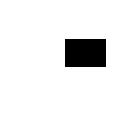


**Exercise 1: Add the white Gaussian noise to coins.png, and then use addition to de-noise. (Adding 100 times first, then using mean). Request to show the original, original with noise, and de-noised images.**

**Exercise 2: Generate images A, and B first. Use “and”, “or”, “not” to get image C, D and E.**

A B

**C D E (not A)**