Experiment No.: 05

Experiment Name: Experimental Study & Analysis of Different Types of Transformation Using MATLAB Software.

Objectives:

- To know about different types transformation of images.
- To know about Fourier Spectrum, Centered Spectrum and Spectrum by log transform.
- To know about image Projective, Affine, Similarity.

5.1 Theory:

Fourier analysis is used in image processing in much the same way as with one-dimensional signals. However, images do not have their information encoded in the frequency domain, making the techniques much less useful. For example, when the Fourier transform is taken of an audio signal, the confusing time domain waveform is converted into an easy-to-understand frequency spectrum. In comparison, taking the Fourier transform of an image converts the straightforward information in the spatial domain into a scrambled form in the frequency domain. In short, don't expect the Fourier transform to help you understand the information encoded in images.
Likewise, don't look to the frequency domain for filter design. The basic feature in images is the edge, the line separating one object or region from another object or region. Since an edge is composed of a wide range of frequency components, trying to modify an image by manipulating the frequency spectrum is generally not productive. Image filters are normally designed in the spatial domain, where the information is encoded in its simplest form. Think in terms of smoothing and edge enhancement operations (the spatial domain) rather than high-pass and low-pass filters (the frequency domain).
In spite of this, Fourier image analysis does have several useful properties. For instance, convolution in the spatial domain corresponds to multiplication in the frequency domain. This is important because multiplication is a simpler mathematical operation than convolution. As with one-dimensional signals, this property enables FFT convolution and various deconvolution techniques. Another useful property of the frequency domain is the Fourier Slice Theorem, the relationship between an image and its projections (the image viewed from its sides). This is the basis of computed tomography, an x-ray imaging technique widely used medicine and industry.
The frequency spectrum of an image can be calculated in several ways, but the FFT method presented here is the only one that is practical. The original image must be composed of N rows by N columns, where N is a power of two, i.e., 256, 512, 1024, etc. If the size of the original image is not a power of two, pixels with a value of zero are added to make it the correct size. We will call the two-dimensional array that holds the image the real array. In addition, another array of the same size is needed, which we will call the imaginary array.

5.2 Equipment:

- Computer
- MATLAB Software
- Images

5.3 Problems:

i. Take an Image and find its Fourier Spectrum, Centered Spectrum and Spectrum by Log Transform.

Code:

```
clc
clear
close all
I = imread('Solid.tif');
subplot(2,2,1)
imshow(I)
title("Original Photo")
F = abs(fft2(I));
F = (F.*255)./max(F);
Fc = fftshift(F);
logt = log(1 + abs(Fc));
subplot(2,2,2)
imshow(F)
title("Fourier Spectrum")
subplot(2,2,3)
imshow(Fc)
title("Centered Spectrum")
subplot(2,2,4)
imshow(logt)
title("Spectrum by log transformation")
```

Output:

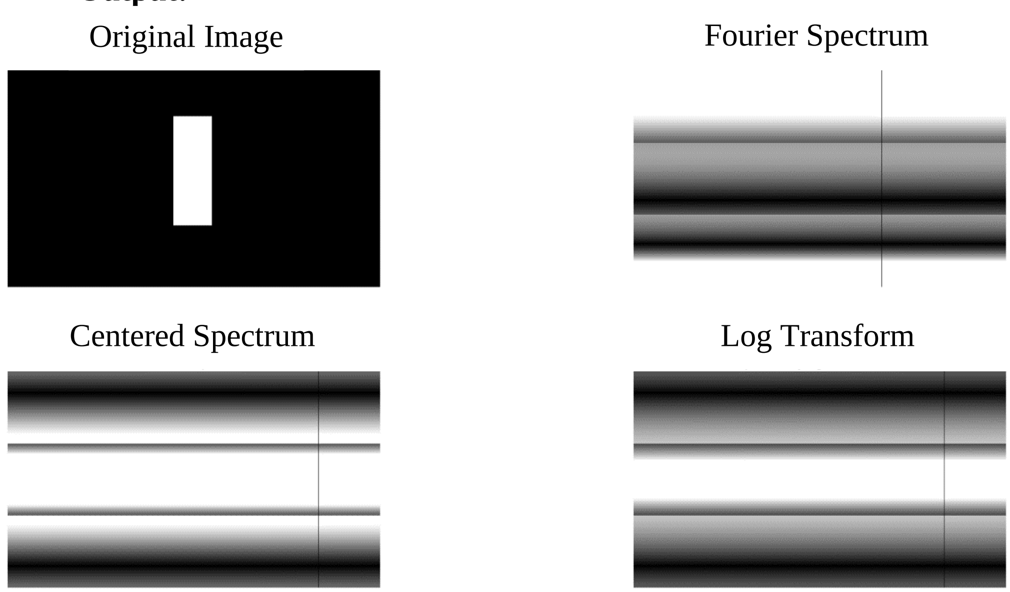

Fig. 5.1: Image Low Pass Filtering

ii. Take an Image and find its transform (Projective, Affine, Similarity).

Code:

```
clc;
clear
sqsize = 60;
I = checkerboard(sqsize,4,4);
nrows = size(I,1);
ncols = size(I,2);
fill = 0.3;
subplot(2,2,1)
imshow(I)
title('Original')

scale = 1.5;        % scale factor
angle = 10*pi/180; % rotation angle
tx = 0;            % x translation
ty = 0;            % y translation
a = -1;            % -1 -> reflection, 1 -> no reflection
sc = scale*cos(angle);
ss = scale*sin(angle);
T = [   sc   -ss  0;
      a*ss  a*sc  0;
        tx    ty  1];
t_sim = affine2d(T);
I_similarity = imwarp(I,t_sim,'FillValues',fill);
subplot(2,2,2)
imshow(I_similarity)
title('Similarity')

T = [1  0.3  0;
     1    1  0;
     0    0  1];
t_aff = affine2d(T);
I_affine = imwarp(I,t_aff,'FillValues',fill);
subplot(2,2,3)
imshow(I_affine)
title('Affine')
T = [1  0  0.002;
     1  1  0.0002;
     0  0  1   ];
t_proj = projective2d(T);
I_projective = imwarp(I,t_proj,'FillValues',fill);
subplot(2,2,4)
imshow(I_projective)
title('Projective')
```

Output:

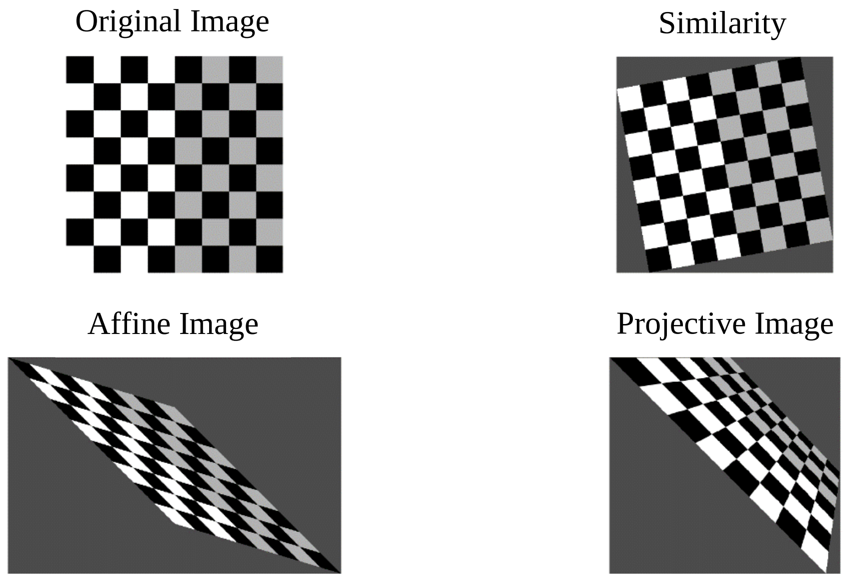

Fig. 5.2: Image Low Pass Filtering

5.4 Conclusion:

In this experiment, we have worked with different Image Spectrum Transform such as Fourier Spectrum, Centered Spectrum and Spectrum by Log Transform. Then we worked with image transform (Projective, Affine, Similarity). All the outputs were as expected, thus the experiment was done successfully.