Digital Image Processing: 525 U0920 / ESOE 5096

Computer Assignment 2
Fourier Transform and Filtering in the Frequency Domain

Due: the week after next, 9:00pm

*Hand in online via **CEIBA***

Total score: 120

This assignment will introduce you to the use of the Fourier transform, which is of fundamental importance to image processing. We will also introduce some filtering techniques in the frequency domain for image enhancement.

What to turn in:

Matlab source codes that perform each individual experiment and the associated results and comments if any. Please consolidate into one single file.

1. **Fourier Transforms of Images** (40%)

First, produce a simple image consisting of a single edge using the following command:

```
img = [zeros(256, 128) ones(256, 128)];
```

We can take its discrete Fourier transform (DFT) via the fast Fourier transform (FFT) algorithm using

```
imgfft = fft2(img);
```

The result should consist of real and imaginary components. To move the origin to the center of the image, we can use

```
fftshift(imgfft);
```

For display purpose, it is often to modify F(u, v) using

```
log(1 + abs(F(u, v)));
```

to reduce the difference between the DC component and others. Please show me the (a) original image, (b) real part, (c) imaginary part, and (d) spectrum of the image after taking the Fourier transform.

Repeat the process for a box:

```
img = zeros(256, 256);
img(78:178, 78:178) = 1;
```

a 45° rotated box:

```
[x, y] = meshgrid(1:256, 1:256);
img= (x+y<329) & (x+y>182) & (x-y>-67) & (x-y<73);
```

and a circle:

```
[x, y] = meshgrid(-128:127, -128:127);
z = sqrt(x.^2 + y.^2);
img = (z < 20);
```

2. Ideal Lowpass and Highpass Filtering (40%)

Read the *cameraman.tif* file and take its Fourier transform using the techniques illustrated in Problem 1. Now, we can perform an ideal lowpass filter by multiplying the transformed matrix by the circle matrix in Problem 1 using

```
imglp = imgfft .* cir;
```

where cir represents the circle matrix. After lowpass filtering with a cutoff frequency of **5**, we can take the inverse Fourier transform to convert it back to the spatial domain using

```
ifft2(imglp);
```

Show me the (a) original image, (b) spectrum, (c) spectrum after lowpass filtering, and (d) lowpass filtered image. Please comment on the results.

Repeat the process using a cutoff frequency of **30**.

Repeat the process using an ideal highpass filter instead for both cutoff frequencies.

3. Gaussian Filtering (40%)

Read the *lena.bmp* file and take its Fourier transform. We can use the following function to create a Gaussian filter:

```
fspecial(…);
```

Note that you can set the filter size equal to the image size. Use two different sigma values of **10** and **30** to create lowpss filters and apply them to the transformed image. Finally, show me the (a) original image, (b) spectrum, (c) spectrum after lowpass filtering, and (d) lowpass filtered image for each individual sigma value. Please comment on the results.

Repeat the process using highpass filters instead for both sigma values.