CMP9135M – Computer Vision

Workshop – Fourier Analysis & Co-occurrence Matrix

Outline & Aim:

Recent lectures presented the concepts of Fourier Analysis, and the estimation of co-occurrence matrices. The aim of this workshop is to implement and familiarise with those two techniques.

Task 1

Open Matlab and create a new Script file named Fourier_Analysis.m. Type the following code, which creates an image and then applies Fourier Transform.

```
% Prepare image
f = zeros(256,256);
f(43:213,111:146) = 1;
imshow(f);

% Compute Fourier Transform
F = fft2(f,256,256);
figure;
imshow(F);
F = fftshift(F); % Center FFT
figure;
imshow(F);

% Measure the minimum and maximum value of the transform amplitude
min(min(abs(F)))
max(max(abs(F)))
figure;
imshow(abs(F),[0 100]); colormap(jet); colorbar
figure;
imshow(log(1+abs(F)),[0,3]); colormap(jet); colorbar


% Look at the phases
figure;
imshow(angle(F),[-pi,pi]); colormap(jet); colorbar
```

What is the main difference between representing the amplitude and its logarithm?

Add the following code at the end of your current file, and then execute

```

I = ifft2(F, 256, 256);
```

```
figure;
imshow(I);
```

Which is the result and why?

Now replace the previous three lines of code with:

```
F = fft2(f,256,256);
figure;
imshow(F);

I = ifft2(F, 256, 256);
figure;
imshow(I);
```

Which is the result and why?

Repeat the previous steps using the images, Test_1, Test_2, Test_3, Test_4 and Gray_21 images. Take care that you convert the colour images to grey scale and that you change the size accordingly. It may be worth to rotate the images (in particular Gray_21) by 90° and observe what happens.

```
Result = imrotate(Image,angle);
```

Which are your observations?

Task 2

Open Matlab and create a new Script file named Co_Occurence.m. Type the following code, which reads an image and then estimates its co-occurrence matrix.
If you use Test_1 and Test_2 images then convert them to greyscale:

```
f = imread('Test_1.tiff');
imshow(f);
f = rgb2gray(f);
figure;
imshow(f);
```

```
f = imread('Test_1.tiff');
imshow(f);
f = rgb2gray(f);
figure;
imshow(f);

glcm_0_s = graycomatrix(f, 'offset', [0 1],'NumLevels', 256, 'Symmetric',
true);
figure;
imshow(glcm_0_s);
glcm_0_ns = graycomatrix(f, 'offset', [0 1], 'NumLevels', 256,'Symmetric',
false);
figure;
imshow(glcm_0_ns);
```

Modify the code provided so that you estimate the co-occurrence matrices not only for 0°, but for 45°, 90°, and 135°, as well as the average matrix. This is done by changing the ‘offset’ parameter as follows:

Angle	Offset
0	[0 D]
45	[-D D]
90	[-D 0]
135	[-D -D]

In addition you can change the parameter D so instead of 1 (i.e. the neighbouring pixel), to be 2, 3 and 4.

After familiarising with this, try to estimate some parameters of the co-occurrence matrix, using the code:

```
f = imread('Test_1.tiff');
imshow(f);
f = rgb2gray(f);
figure;
imshow(f);

glcm_0_s = graycomatrix(f, 'offset', [0 1],'NumLevels', 256, 'Symmetric', true);
figure;
imshow(glcm_0_s);
glcm_0_ns = graycomatrix(f, 'offset', [0 1], 'NumLevels', 256,'Symmetric', false);
figure;
imshow(glcm_0_ns);

stats_0_s = graycoprops(glcm_0_s,{'contrast','correlation',
```

```
'energy','homogeneity'})

stats_0_ns = graycoprops(glcm_0_ns,{'contrast','correlation',
'energy','homogeneity'})
```

Are the estimated parameters the same or different? Why is that?

As an additional task, try to rotate the input image, and rerun the above code. Does orientation and/or image size have an effect?

END OF WORKSHOP