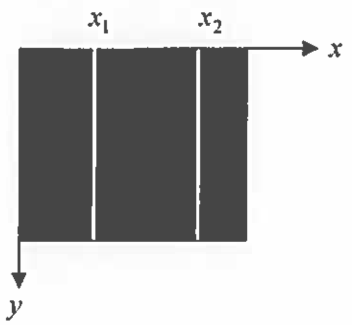
1. (a)

(i) Plot the image intensity.



[1]

(ii) For an image which contains only a single non-zero edge at $x = x_1$, the $M \times N$-point Discrete Fourier Transform (DFT) of $f(x, y)$ is given as follows:

$$F(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M+vy/N)} = \frac{1}{MN}\sum_{y=0}^{N-1} f(x_1,y)e^{-j2\pi(ux_1/M+vy/N)}$$

$$= \frac{1}{MN}ce^{-j2\pi ux_1/M}\sum_{y=0}^{N-1} e^{-j2\pi vy/N} = \frac{1}{MN}ce^{-j2\pi ux_1/M}\frac{1-(e^{-j2\pi vy/N})^N}{1-e^{-j2\pi vy/N}}$$

$$= \frac{1}{MN}ce^{-j2\pi ux_1/M}\frac{1-e^{-j2\pi vy}}{1-e^{-j2\pi vy/N}}$$

$$F(u,v) = \begin{cases} \frac{1}{M}ce^{-j2\pi ux_1/M}, & v = 0 \\ 0, & \text{otherwise} \end{cases}$$

For the image with 2 non-zero edges

$$F(u,v) = \begin{cases} \frac{1}{M}c(e^{-j2\pi ux_1/M} + e^{-j2\pi ux_2/M}), & v = 0 \\ 0, & \text{otherwise} \end{cases}$$

[5]

(iii) As seen a set of parallel straight lines in space implies a straight line perpendicular to the original one in frequency. [2]

(b)

(i) The 1-D Hadamard transform kernel is defined as

$H(x,u) = \prod_{i=0}^{n-1}(-1)^{b_i(x)b_i(u)}$. For signals of size 2 samples the Hadamard matrix is 2x2 and the Hadamard kernel has 4 samples as follows:

$$H(x,u) = \prod_{i=0}^{1-1}(-1)^{b_i(x)b_i(u)} = (-1)^{b_0(x)b_0(u)}$$

$$H(0,0) = (-1)^{0\cdot 0} = 1$$

$$H(0,1) = (-1)^{0\cdot 1} = 1$$

$$H(1,0) = (-1)^{1\cdot 0} = 1$$

$$H(1,1) = (-1)^{1\cdot 1} = -1$$

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Using the recursive relationship of the Hadamard matrix we get:

$$H_4 = \begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

We apply this matrix in the given image row-by-row and column-by-column or the other way round. We obtain:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} r \\ r \\ r \\ r \end{bmatrix} = \begin{bmatrix} 4r \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Therefore, the intermediate image is:

$\begin{bmatrix} 4r & 0 & 0 & 0 \\ 4r & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$. Then

$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 4r \\ 4r \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 8r \\ 0 \\ 8r \\ 0 \end{bmatrix}$. Therefore, the final image is:

$$\begin{bmatrix} 8r & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 8r & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

[5]

(c)

(i) Values of a, b, c must be real and positive.

The eigenvalues of the covariance matrix $\underline{C}_f$ are obtained by solving the equation $\det\left|\underline{C}_f - \lambda I\right| = 0 \Rightarrow (a-\lambda)^3 - b^2(a-\lambda) - c^2(a-\lambda) = 0 \Rightarrow \lambda = a, \quad \lambda = a \pm \sqrt{b^2 + c^2}$.

The eigenvalues are sorted as follows: $a + \sqrt{b^2 + c^2} \geq a \geq a - \sqrt{b^2 + c^2}$.

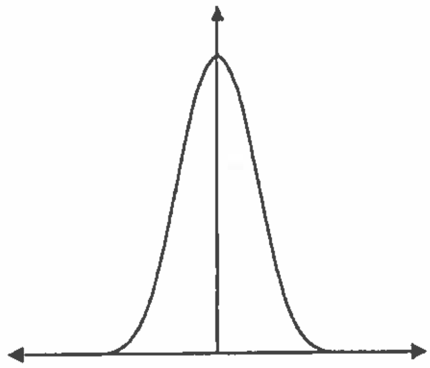
[5]

(ii) If we keep one image the error will be $2a - \sqrt{b^2 + c^2}$ and if we keep two images the error will be $a - \sqrt{b^2 + c^2}$. **[2]**

2. (a)

(i) The new histogram will be the convolution of the 2 original histograms and therefore it will occupy a wider range of values. Therefore, the new image will be an image of higher contrast compared to the original. **[2]**

(ii) By carrying out the proposed manipulation we see that most pixel values of pixels which belong to a constant or slowly varying area will turn into zeros. Furthermore, the resulting intensities will be both positive and negative. The resulting histogram will look as follows:



[2]

(b)

Large spatial masks can be used within flat or slowly varying areas whereas small masks can be used within high activity areas. The idea behind this approach is that noise is more visible in the first type of areas. Local variance can classify each pixel into the right area. **[2]**

(c)

Bandpass filters are useful for removing background noise without completely eliminating the background information. A bandpass filter can be implemented by a spatial mask as follows. The original image $f(x,y)$ is first convolved with a spatial mask of size $N_1 \times N_1$ to produce an output $g_1(x,y)$. Then it is convolved with a spatial mask of size $N_2 \times N_2$ with $N_2 > N_1$ to produce an output $g_2(x,y)$. The final output is obtained as the difference $g_1(x,y) - g_2(x,y)$ and this is the bandpass filtered version of the image. **[3]**

(d)

It can easily be deducted that a pair of mask which detect edges at 45 and -45 degrees could be the following:

1	1	0
1	0	-1
0	-1	-1

-1	-1	0
-1	0	1
0	1	1

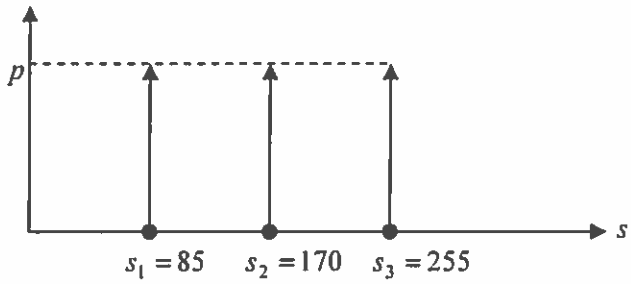
[3]

(e)

(i) $f(x,y)$ is dark image since the intensities are concentrated in the lower half of the intensity range. Moreover, it consists of three intensities only with equal probabilities, and therefore, $p=\frac{1}{3}$. **[2]**

(ii) We can use histogram equalization. By doing so, the three intensities are mapped to the following: $s_1 = T(r_1) = \frac{1}{3}255 = 85$, $s_2 = T(r_2) = \frac{2}{3}255 = 170$, $s_3 = T(r_3) = 255$ **[2]**

(iii)



[2]

(iv) For the original image we have:

Mean: $m_1 = \frac{1}{3}(r_1 + r_2 + r_3)$

Variance: $\sigma_1^2 = \frac{1}{3}(r_1^2 + r_2^2 + r_3^2) - \frac{1}{9}(r_1 + r_2 + r_3)^2$

For the equalised image we have:

Mean: $m_2 = 170$

Variance: $\sigma_2^2 = \frac{1}{3}(s_1^2 + s_2^2 + s_3^2) - 170 \cdot 170 = \frac{14450}{3}$ **[2]**

3. (a)

(i) For a very small $k_2(x,y)$ the factor $e^{-k_2(x,y)(m^2+n^2)}$ tends to 1 and therefore, we can write $h_{(x,y)}(m,n) = k_1(x,y)w(m,n)$. **[3]**

(ii) For a very large $k_2(x,y)$ we can write

$$e^{-k_2(x,y)(m^2+n^2)} = \begin{cases} 1 & m = n = 0 \\ 0 & \text{otherwise} \end{cases}$$

Therefore, we can write

$h_{(x,y)}(m,n) = k_1(x,y)\delta(m,n)$ **[3]**

(iii) In image regions of high detail, such as edge regions, we don't want to do any filtering and therefore we want large k_2. In image regions of low detail, such as uniform background regions, we want to de-noise and therefore we want small k_2. Therefore, we can choose $k_2(x,y) = \sigma_g^2(x,y)$ **[3]**

(iv) We wish the sum of the filter coefficients to be 1 so that we don't distort the mean intensity of the image, and therefore,

$$k_1(x,y) = \frac{1}{\sum_{m=-2}^{2}\sum_{n=-2}^{2} e^{-\sigma_g^2(x,y)(m^2+n^2)}}$$ **[3]**

(v) A possible scenario is to filter both high detail and bright areas. Bright areas can be identified from the local mean m_g (A large m_g indicates a bright area). Therefore, we can normalize σ_g^2 and m_g so that they occupy the same range of values (for example from 0 to 1) to obtain $\tilde{\sigma}_g^2$ and $\tilde{m}_g$ and then we can consider $k_2 = \tilde{\sigma}_g^2 + \tilde{m}_g$. **[3]**

(b)

(i) CLS restoration refers to the solution of the following constrained minimization problem.

Minimize: $J(\mathbf{f}) = \|\mathbf{n}(\mathbf{f})\|^2 = \|\mathbf{y} - \mathbf{Hf}\|^2$

Subject to: $\|\mathbf{Cf}\|^2 < \varepsilon$

where $\mathbf{Cf}$ is a high pass filtered version of the image.

The idea behind the above constraint is that the highpass version of the image contains a considerably large amount of noise.

The minimization of the above leads to the following estimate for the original image

$$\mathbf{f} = \left(\mathbf{H}^T\mathbf{H} + \alpha\mathbf{C}^T\mathbf{C}\right)^{-1}\mathbf{H}^T\mathbf{y}$$

where α is the so called regularization parameter that controls the contribution between the two terms in the cost function. **[2]**

(ii) For large regularization parameter, we achieve strong elimination of high frequencies. This is desirable if the image of interest is heavily corrupted by noise. For small regularization parameter the CLS restoration tends to imitate Inverse Filtering. **[3]**

4. (a)

(i) The three reconstruction levels are:

$s_1 = \frac{1}{2}\frac{255}{3} \cong 43$, $s_2 = 43 + 85 = 128$, $s_3 = 128 + 43 = 171$. **[5]**

(ii) The equation which describes the pdf function is of the form $y = \frac{2}{255 \cdot 255}x$. Based on this we can find the probabilities of the 3 reconstruction levels.

The probability of s_1 is $\frac{1}{2}\frac{255}{3}\frac{2}{255 \cdot 255}\frac{255}{3} = \frac{1}{9}$. The probability of s_2 is $\frac{1}{2}\frac{2 \cdot 255}{3}\frac{2}{255 \cdot 255}\frac{2 \cdot 255}{3} - \frac{1}{9} = \frac{3}{9} = \frac{1}{3}$. The probability of s_3 is $\frac{5}{9}$.

Symbol	Probability	Code
$s_1 = 43$	1/9	11
$s_2 = 128$	3/9	10
$s_3 = 171$	5/9	0

Average number of bits per symbol $l_{avg} = \frac{1}{9} \cdot 2 + \frac{3}{9} \cdot 2 + \frac{5}{9} \cdot 1 = 1.444$ bits/symbol.

[5]

(iii) Entropy $H(s) = n = 1.35164$ bits/symbol.

Redundancy $l_{avg} - n = 1.444 - 1.35164 = 0.09236$ or $\frac{l_{avg} - n}{n}\% = 6\%$ of entropy. Huffman code exhibits a low redundancy for the specific alphabet and therefore is efficient enough. **[5]**

(b)

$a = 144$, $b = 191$, $c = 190$ and $x = 180$. Let $y = a + b - c = 144 + 191 - 190 = 145$; then $y = 145$, and the prediction residual is $r = 145 - 180 = -35$. -35 belongs to category 6. The binary number for 35 is 100011, and its one's complement is 011100. Thus, -35 is represented as (6,011100). If the Huffman code for six is 1110, then -35 is coded by the 10-bit codeword 1110011100. Without entropy coding, -35 would require 16 bits. **[5]**

p 6