DIP Exam 2021 solutions

1. (a) Let $f(x,y)$ denote an $N \times N$ −point two-dimensional sequence, that has zero value outside $0 \leq x \leq N-1$, $0 \leq y \leq N-1$, where N is an integer power of 2. In implementing the standard Walsh Transform of $f(x,y)$, we relate $f(x,y)$ to a new $N \times N$ −point sequence $W(u,v)$.

(i) Define the sequence $W(u,v)$ in terms of $f(x,y)$. Explain all the symbols used. **[1]**

Answer

$$W(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)(-1)^{\sum_{i=0}^{n-1}(b_i(x)b_{n-1-i}(u)+b_i(y)b_{n-1-i}(v))}$$

We use binary representation for the values of the independent variables x and u. We need n bits to represent them, $N = 2^n$. Hence, for the binary representation of x and u we can write:

$(x)_{10} = (b_{n-1}(x)b_{n-2}(x) \dots b_0(x))_2$

$(u)_{10} = (b_{n-1}(u)b_{n-2}(u) \dots b_0(u))_2$ with $b_i(x)$ 0 or 1 for $i = 0, \dots, n-1$.

(ii) For the case $N = 2$ and $f(x,y) = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$ calculate the forward Walsh transform coefficients. **[4]**

Answer

We know that $N = 2^n$ and therefore, in case of $N = 2$ we have $n = 1$, x, y 0 or 1 and $b_0(0) = 0$, $b_0(1) = 1$. For $f(x,y) = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$ we calculate the Hadamard transform coefficients as follows.

$$H(u,v) = \frac{1}{N}\sum_{x=0}^{2-1}\sum_{y=0}^{2-1} f(x,y)\left[\prod_{i=0}^{n-1}(-1)^{(b_i(x)b_{n-1-i}(u)+b_i(y)b_{n-1-i}(v))}\right]$$

$$H(u,v) = \frac{1}{N}\sum_{x=0}^{1}\sum_{y=0}^{1} f(x,y)\left[\prod_{i=0}^{1-1}(-1)^{(b_i(x)b_{-i}(u)+b_i(y)b_{-i}(v))}\right]$$

$$H(u,v) = \frac{1}{N}\sum_{x=0}^{1}\sum_{y=0}^{1} f(x,y)(-1)^{(b_0(x)b_0(u)+b_0(y)b_0(v))}$$

$$= \frac{1}{2}f(0,0)(-1)^{(b_0(0)b_0(u)+b_0(0)b_0(v))} + \frac{1}{2}f(0,1)(-1)^{(b_0(0)b_0(u)+b_0(1)b_0(v))}$$
$$+ \frac{1}{2}f(1,0)(-1)^{(b_0(1)b_0(u)+b_0(0)b_0(v))} + \frac{1}{2}f(1,1)(-1)^{(b_0(1)b_0(u)+b_0(1)b_0(v))}$$
$$= \frac{1}{2}f(0,0)(-1)^{(0\cdot b_0(u)+0\cdot b_0(v))} + \frac{1}{2}f(0,1)(-1)^{(0\cdot b_0(u)+1\cdot b_0(v))}$$
$$+ \frac{1}{2}f(1,0)(-1)^{(1\cdot b_0(u)+0\cdot b_0(v))} + \frac{1}{2}f(1,1)(-1)^{(1\cdot b_0(u)+1\cdot b_0(v))}$$
$$= \frac{1}{2}f(0,0)(-1)^0 + \frac{1}{2}f(0,1)(-1)^{b_0(v)} + \frac{1}{2}f(1,0)(-1)^{b_0(u)}$$
$$+ \frac{1}{2}f(1,1)(-1)^{b_0(u)+b_0(v)}$$
$$= \frac{3}{2}(-1)^0 + \frac{1}{2}(-1)^{b_0(v)} + \frac{1}{2}(-1)^{b_0(u)} + \frac{2}{2}(-1)^{b_0(u)+b_0(v)}$$
$$= \frac{3}{2} + \frac{1}{2}(-1)^{b_0(v)} + \frac{1}{2}(-1)^{b_0(u)} + (-1)^{b_0(u)+b_0(v)}$$

$$H(0,0) = \frac{3}{2} + \frac{1}{2}(-1)^{b_0(0)} + \frac{1}{2}(-1)^{b_0(0)} + (-1)^{b_0(0)+b_0(0)}$$
$$= \frac{3}{2} + \frac{1}{2}(-1)^0 + \frac{1}{2}(-1)^0 + (-1)^0 = \frac{3}{2} + \frac{1}{2} + \frac{1}{2} + 1 = \frac{7}{2}$$
$$H(0,1) = \frac{3}{2} + \frac{1}{2}(-1)^{b_0(0)} + \frac{1}{2}(-1)^{b_0(1)} + (-1)^{b_0(0)+b_0(1)}$$
$$= \frac{3}{2} + \frac{1}{2}(-1)^0 + \frac{1}{2}(-1)^1 + (-1)^{0+1} = \frac{3}{2} + \frac{1}{2} - \frac{1}{2} - 1 = \frac{1}{2}$$
$$H(1,0) = \frac{3}{2} + \frac{1}{2}(-1)^{b_0(1)} + \frac{1}{2}(-1)^{b_0(0)} + (-1)^{b_0(1)+b_0(0)}$$
$$= \frac{3}{2} + \frac{1}{2}(-1)^1 + \frac{1}{2}(-1)^0 + (-1)^{1+0} = \frac{3}{2} - \frac{1}{2} + \frac{1}{2} - 1 = \frac{1}{2}$$
$$H(1,1) = \frac{3}{2} + \frac{1}{2}(-1)^{b_0(1)} + \frac{1}{2}(-1)^{b_0(1)} + (-1)^{b_0(1)+b_0(1)}$$
$$= \frac{3}{2} + \frac{1}{2}(-1)^1 + \frac{1}{2}(-1)^1 + (-1)^{1+1} = \frac{3}{2} - \frac{1}{2} - \frac{1}{2} + 1 = \frac{3}{2}$$

Therefore, $H(u,v) = \begin{bmatrix} \frac{7}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{3}{2} \end{bmatrix}$.

(b) Consider the population of vectors $\underline{f}$ of the form

$$\underline{f}(x,y) = \begin{bmatrix} f_1(x,y) \\ f_2(x,y) \\ f_3(x,y) \end{bmatrix}.$$

Each component $f_i(x,y),\ i = 1,2,3$ represents an image of size $M \times M$ where M is even. The population arises from the formation of the vectors $\underline{f}$ across the entire collection of pixels (x,y).

Consider now a population of vectors of the form

$$\underline{g}(x,y) = \begin{bmatrix} g_1(x,y) \\ g_2(x,y) \\ g_3(x,y) \end{bmatrix}$$

where the vectors $\underline{g}$ are the Karhunen-Loeve (KL) transforms of the vectors $\underline{f}$.

(i) Prove and explain the relationship between the covariance matrix of $\underline{g}(x,y)$ and the covariance matrix of $\underline{f}(x,y)$. What is the structure of the covariance matrix of $\underline{g}(x,y)$?

[3]

Answer

Using the relationships

$$\underline{g} = \underline{A}(\underline{x} - \underline{m_x})$$
$$\underline{g}^T = [\underline{A}(\underline{x} - \underline{m_x})]^T = (\underline{x} - \underline{m_x})^T \underline{A}^T$$

we get

$$\underline{g}\underline{g}^T = \underline{A}(\underline{x} - \underline{m_x})(\underline{x} - \underline{m_x})^T \underline{A}^T \Rightarrow E\{\underline{g}\underline{g}^T\} = E\{\underline{A}(\underline{x} - \underline{m_x})(\underline{x} - \underline{m_x})^T \underline{A}^T\}$$
$$= \underline{A}E\{(\underline{x} - \underline{m_x})(\underline{x} - \underline{m_x})^T\}\underline{A}^T$$

$$\underline{C_g} = \underline{A}\,\underline{C_x}\underline{A}^T$$

$$\underline{C_x}\underline{A}^T = \underline{C}_x[\underline{e_1} \quad \underline{e_2} \quad \cdots \quad \underline{e_n}] = [\lambda_1\underline{e_1} \quad \lambda_2\underline{e_2} \quad \cdots \quad \lambda_n\underline{e_n}]$$

$$\underline{C_g} = \underline{AC_xA}^T = \begin{bmatrix} \underline{e_1}^T \\ \underline{e_2}^T \\ \vdots \\ \underline{e_n}^T \end{bmatrix} [\lambda_1\underline{e_1} \quad \lambda_2\underline{e_2} \quad \cdots \quad \lambda_n\underline{e_n}]$$

Because $\underline{e_i}$ is a set of orthonormal eigenvectors we have that:

$$\underline{e_i^T}\,\underline{e_i} = 1, i = 1, \dots, n$$

$\underline{e_i^T}\,\underline{e_j} = 0, i, j = 1, \dots, n$ and $i \neq j$
and therefore, $\underline{C_g}$ is a diagonal matrix whose elements along the diagonal are the eigenvalues of $\underline{C_x}$

(ii) Can the elements of the covariance matrix of $\underline{g}(x, y)$ be negative? Justify your answer. **[2]**

Answer

They cannot be negative since they represent variances.

(iii) Suppose that $N = 8$ and that the eigenvalues of the covariance matrix of $\underline{f}(x, y)$ are:

$$[168 \quad 6.1 \quad 0.08 \quad 13 \quad 64 \quad 214 \quad 1.2 \quad 0.2]$$

What will be the mean square error if we use principal component images associated with the largest eigenvalues for 2: 1 and 4: 1 data compression? **[4]**

Answer

The mean square error for compression 2: 1 is $\sum_{j=5}^{8} \lambda_j = 7.58$.
The mean square error for compression 4: 1 is $\sum_{j=3}^{8} \lambda_j = 84.58$.

(c) Consider again Question 1(b) in the case of 2 images ($N = 2$). The covariance matrix of the population is $\underline{C_f}$ with elements defined as $C_{m,n} = \rho^{|m-n|}, 1 \leq m, n \leq 2, 0 < \rho < 1$.

(i) What are the variances of the images $g_1(x, y)$ and $g_2(x, y)$? **[4]**

Answer

The covariance matrix of $\underline{f}$ is $\begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ and the eigenvalues of this matrix which are the variances of the images $g_1(x, y)$ and $g_2(x, y)$ are $\lambda_1 = 1 + \rho, \lambda_2 = 1 - \rho$.

(ii) What will be the mean square error as a function of ρ if we use the principal component image from the set of images $g_1(x, y)$ and $g_2(x, y)$ to reconstruct the original images? **[2]**

Answer

The error is $1 - \rho$

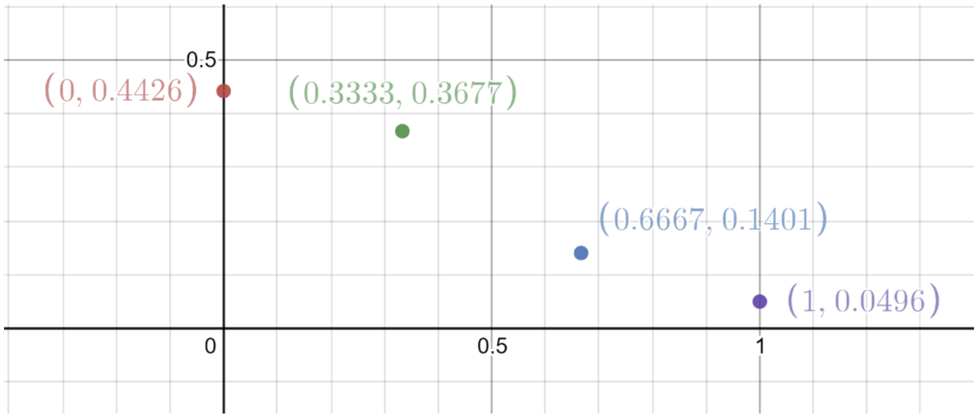
2. (a) Consider a 64×64 image with 4 grey levels. The normalized grey levels are denote an $0, \frac{1}{3}, \frac{2}{3}$ and 1. The number of pixels with the corresponding grey levels, are shown in the following table.

Grey level	Number of pixels
0	1813
$\frac{1}{3}$	1506
$\frac{2}{3}$	574
1	203

(i) Draw the histogram of the image. **[1]**

Answer

$p(0) = \frac{1813}{4096} = 0.4426$, $p\left(\frac{1}{3}\right) = \frac{1506}{409} = 0.3677$, $p\left(\frac{2}{3}\right) = \frac{574}{4096} = 0.1401$, $p(1) = \frac{203}{4096} = 0.0496$.



(ii) Determine the equalized histogram. **[2]**

Answer

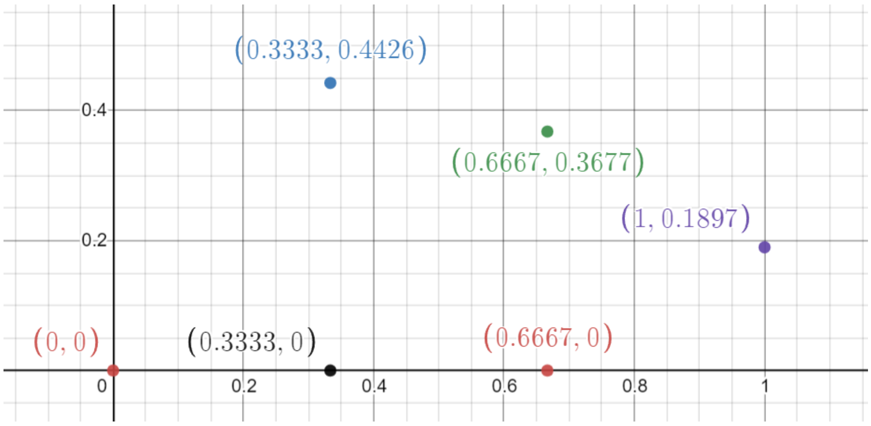
What we provide below is the transformation of each original intensity. The mappings yield real values which must be allocated to the nearest value among $\{0, \frac{1}{3}, \frac{2}{3}, 1\}$.

$T(0) = \frac{1813}{4096} = 0.4426 \rightarrow \mathbf{1/3}$, $T\left(\frac{1}{3}\right) = \frac{1506+1813}{4096} = 0.8103 \rightarrow \mathbf{2/3}$,
$T\left(\frac{2}{3}\right) = \frac{574+1506+1813}{4096} = 0.9504 \rightarrow \mathbf{1}$, $T(1) = \mathbf{1}$.

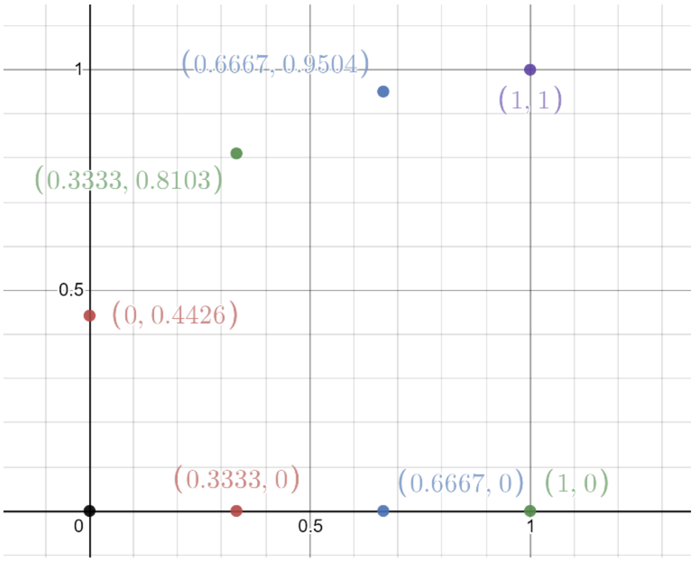
(iii) Draw the equalized histogram. **[1]**

Answer

In the equalised histogram the intensity 0 does not appear.
The probability of 1/3 is equal to the probability of 0 in the original image, i.e., 0.4426.
The probability of 2/3 is equal to the probability of 1/3 in the original image, i.e., 0.3677.
The probability of 1 is equal to the sum of probabilities of 2/3 and 1 in the original image, i.e., 0.1897.



The transformation is shown below:



(b) What happens if you apply histogram equalisation twice to the same image? Justify your answer. **[4]**

Answer

By applying histogram equalisation to an image with intensity r, that ranges from 0 to $L-1$, a new image with intensity s is produced as follows:
$s = T(r) = (L-1)\int_0^r p_r(w)dw, 0 \leq r \leq L-1$
with $p_s(s) = \frac{1}{L-1}, \ 0 \leq s \leq L-1$
If we apply histogram equalisation to the new image s we obtain image an image t as follows:
$$t = T(s) = (L-1)\int_0^s p_s(w)dw = (L-1)\frac{s}{L-1} = s$$
We see that $t = s$ and therefore, applying histogram equalisation twice makes no sense.

(c) After histogram equalisation will an image have more, the same or fewer distinct grey levels? Justify your answers. **[4]**

Answer

Theoretically, the equalised image will have the same number of distinct grey levels, since each original grey level is mapped into a new distinct grey level. However, due to quantisation, more than one original grey levels might be mapped onto the same new grey level and therefore, in a real-life scenario the equalised image might have fewer distinct grey levels.

(d) A mean filter is a linear filter, but a median filter is not. Why? Justify your answer. **[4]**

Answer

This argument can be demonstrated by simple example, i.e., median$\{0,1,0\}$+median$\{2,0,0\} = 0 + 0 = 0$. This is not the same as the median of $[0,1,0\} + \{2,0,0\} = \{2,1,0\}$ which is 1.

(e) Let $f(x, y)$ denote an $M \times N$ image. Suppose that the pixel intensities r are represented by 8 bits. Moreover, the histogram $h(r)$ of the image is available. Find the value of $\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |F(u, v)|^2$, with $F(u, v)$ with the Discrete Fourier Transform of the image.

[4]

Answer

Due to Parseval's Theorem $\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |F(u, v)|^2 = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)^2$.
Suppose we have a digital image of size $M \times N$ with grey levels in the range $[0, L - 1]$. The histogram of the image is defined as the following discrete function:

$$h(r_k) = n_k$$

where
r_k is the kth grey level, $k = 0,1, \dots ,255$
n_k is the number of pixels in the image with grey level r_k
$\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |F(u, v)|^2 = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)^2 = \sum_{i=0}^{255} {r_i}^2 \, n_i = \sum_{i=0}^{255} {r_i}^2 \, h(r_i) \Rightarrow$

$$\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |F(u, v)|^2 = \sum_{i=0}^{255} {r_i}^2 \, h(r_i)$$

Note that some authors define $h(r_k) = n_k$ and some $h(r_k) = n_k/MN$. Both definitions are acceptable.

3. We are given the degraded version $g(x,y)$ of an image $f(x,y)$ such that in lexicographic ordering

$$g = Hf + n$$

where H is the degradation matrix which is assumed to be block-circulant and n is the noise term which is assumed to be zero-mean, white and independent of the image f. The symbols f, n, g denote the lexicographically ordered original, noise and degraded image respectively. All images involved have size $N \times N$ after extension and zero-padding.

(a) (i) Under what assumptions can we perfectly restore the image using the technique of Inverse Filtering? **[3]**

Answer

In the presence of external noise we have that

$$\hat{F}(u,v) = \frac{H^*(u,v)\big(Y(u,v) - N(u,v)\big)}{|H(u,v)|^2} =$$

$$\frac{H^*(u,v)Y(u,v)}{|H(u,v)|^2} - \frac{H^*(u,v)N(u,v)}{|H(u,v)|^2} \Rightarrow$$

$$\hat{F}(u,v) = F(u,v) - \frac{N(u,v)}{H(u,v)}$$

If $H(u,v)$ becomes very small, the term $N(u,v)$ dominates the result.

(ii) How can we avoid erratic behaviour when the assumption is not met? **[3]**

Answer

We carry out the restoration process in a limited neighborhood about the origin where $H(u,v)$ is not very small. This procedure is called **pseudoinverse filtering**.
In that case we set

$$\hat{F}(u,v) = \begin{cases} \dfrac{H^*(u,v)Y(u,v)}{|H(u,v)|^2} & H(u,v) \neq 0 \\ 0 & H(u,v) = 0 \end{cases}$$

or

$$\hat{F}(u,v) = \begin{cases} \dfrac{H^*(u,v)Y(u,v)}{|H(u,v)|^2} & |H(u,v)| \geq \varepsilon \\ 0 & \text{otherwise} \end{cases}$$

In general, the noise may very well possess large components at high frequencies (u,v), while $H(u,v)$ and $Y(u,v)$ normally will be dominated by low frequency components. The parameter ε (called **threshold**) is a small number chosen by the user.

(iii) Discuss the relation between Wiener filtering and both inverse and pseudo-inverse filtering. **[4]**

Answer

Wiener smoothing filter

In the absence of any blur, $H(u,v)=1$ and

$$W(u,v)=\frac{S_{ff}(u,v)}{S_{ff}(u,v)+S_{nn}(u,v)}=\frac{(SNR)}{(SNR)+1}$$

(a) $(SNR)>>1 \Rightarrow W(u,v)\cong 1$

(b) $(SNR)<<1 \Rightarrow W(u,v)\cong (SNR)$

(SNR) is high in low spatial frequencies and low in high spatial frequencies so $W(u,v)$ can be implemented with a lowpass (smoothing) filter.

Relation with inverse filtering

If $S_{nn}(u,v)=0 \Rightarrow W(u,v)=\dfrac{1}{H(u,v)}$ which is the inverse filter

If $S_{nn}(u,v)\to 0$

$$\lim_{S_{nn}\to 0} W(u,v)=\begin{cases}\dfrac{1}{H(u,v)} & H(u,v)\neq 0\\[2ex] 0 & H(u,v)=0\end{cases}$$

which is the pseudoinverse filter.

(b) Propose a technique to restore an image using a spatially adaptive Constrained Least Squares (CLS) filter. A full mathematical analysis is required. **[4]**

Answer

With larger values of α, and thus more regularisation, the restored image tends to have more ringing.

With smaller values of α, the restored image tends to have more amplified noise effects.

The noise is more visible within constant or slowly varying areas. Therefore, we require, small α close to the edges and large α within constant or slowly varying areas. We might classify each pixel as a background pixel or an edge pixel. If a pixel is a background pixel, we can restore that pixel by defining a rectangular patch around that pixel and restoring the patch using a large α. If a pixel is an edge pixel, we can restore that pixel by defining a rectangular patch around that pixel and restoring the patch using a small α. We can define α $=\frac{1}{\text{local variance}}$. The local variance is small withing a constant or slowly varying area and large close to edges and therefore, using the previous relationship α will be large withing a constant or slowly varying area and large close to edges.

(c) One class of filters considered for reducing background noise in images has frequency response $W(u,v)$ given by:

$$W(u,v)=\left[\frac{S_{ff}(u,v)}{S_{ff}(u,v)+S_{nn}(u,v)}\right]^{\beta}$$

where $S_{ff}(u,v)$ is the original image power spectrum and $S_{nn}(u,v)$ is the noise power spectrum. If $\beta=1$, the filter is a Wiener filter. If $\beta=1/2$, the filter is called a power spectrum filter. Suppose $S_{ff}(u,v)$ has a lowpass character and its amplitude decreases as u and v increase., while $S_{nn}(u,v)$ is approximately constant independent of u and v.

(i) For a given $S_{ff}(u, v)$ and $S_{nn}(u, v)$, which filter among the Wiener filter and the power spectrum filter performs better in term of noise reduction? Justify your answer. **[3]**

Answer

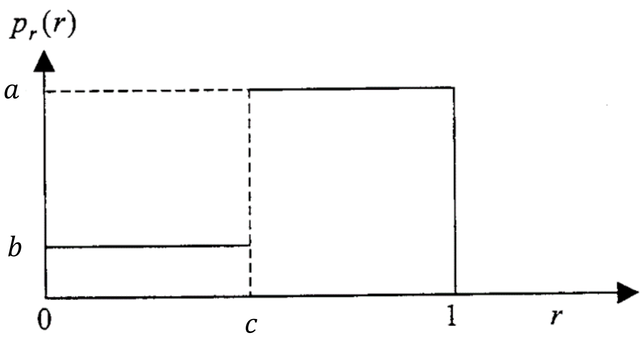
The power $\beta = 1/2$ increases the value of the filter so that power spectrum filter removes lees noise than the Wiener filter.

(ii) For a given $S_{ff}(u, v)$ and $S_{nn}(u, v)$, which filter among the Wiener filter and the power spectrum filter performs worse in term of image blurring? Justify your answer. **[3]**

Answer

According to (i) the Wiener filter.

4. (a) Consider an image with intensity $f(x, y)$ that can be modeled as a sample obtained from the probability density function sketched below:



(i) Suppose that four reconstruction levels are assigned to quantize the intensity $r = f(x, y)$. Determine these reconstruction levels using a uniform quantizer. In the above figure $a = 1.6, b = 0.4$. **[5]**

Answer

It is easy to find that since $p_r(r)$ is a pdf, the area under the curve should be 1. This gives us $c = 0.5$.
Therefore, the four reconstruction levels are 0.125, 0.375, 0.625 and 0.875. In order to find these reconstruction levels, we divide the interval $[0, 1]$ in 4 equal sub-intervals and we obtain the middle value of these 4 sub-intervals.

(ii) Suppose that the four intensity levels found in (i) are to be transmitted using Huffman coding. Find the Huffman codewords. For your codeword assignment, determine the average number of bits required to represent r. **[5]**

Answer

The probability of the reconstruction levels 0.125 and 0.375 is $0.4 \cdot 0.25 = 0.1$ and the probability of the reconstruction levels 0.625 and 0.875 is $1.6 \cdot 0.25 = 0.4$.

0.625	11
0.875	0
0.375	101
0.125	100

The average number of bits is
$l_{\text{avg}} = 0.1 \cdot 3 + 0.1 \cdot 3 + 0.4 \cdot 1 + 0.4 \cdot 2 = 1.8$ bits per symbol

(iii) Determine the entropy, the redundancy and the coding efficiency of the Huffman code for this example.

Answer

Entropy: $H(s) = 1.7219$ bits per symbol
Redundancy: $l_{\text{avg}} - H(s) = 0.0781$
Coding efficiency: $\frac{H(s)}{l_{\text{avg}}} = 0.9566$

(b) A set of images is to be compressed. Each pixel of each image has a value in the range $[0, 3]$, i.e., we require 2 bits/pixel. An image taken from this set is given below. The frequency of occurrence of pixels of different values in this image, is typical of the set of images as a whole.

$$\begin{bmatrix} 3 & 3 & 3 & 2 \\ 2 & 3 & 3 & 3 \\ 3 & 2 & 2 & 2 \\ 2 & 1 & 1 & 0 \end{bmatrix}$$

What is the degree of compression achievable using the following methods? **[4]**

(i) Huffman coding of the pixel values.

Answer

```
Symbol     Probability                                  Huffman Code

  3           7/16                            \0              0
                                                 - 1-
  2           6/16                 \0         /1             10
                                        -9/16-
  1           2/16   \0            /1                        110
                         - 3/16 -
  0           1/16   /1                                      111
```

The average number of bits is

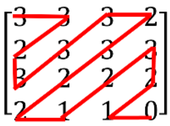
$$l_{\text{avg}} = \frac{2}{16} \cdot 3 + \frac{1}{16} \cdot 3 + \frac{7}{16} \cdot 1 + \frac{6}{16} \cdot 2 = 1.75$$

If we use fixed length symbol representation we need 2 bits per symbol. Therefore the compression ratio is $\frac{2}{1.75} = 1.1428$.

(ii) Forming differences between adjacent pixels (assuming zig-zag scan) and then Huffman coding these differences. For the top-left pixel use the value of 0 in this calculation. **[4]**

Answer

The zig-zag scanning shown below yields the matrix with 15 elements shown below as well.



$$\begin{bmatrix} 0 & 0 & 0 & -1 \\ -1 & 0 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 0 & -1 & -1 & -1 \end{bmatrix}$$

Symbol	Probability	Huffman Code
-1	$\frac{7}{16}$	0
0	$\frac{5}{16}$	11
1	$\frac{4}{16}$	10

The average number of bits is $l_{\text{avg}} = \frac{7}{16} \cdot 1 + \frac{5}{16} \cdot 2 + \frac{4}{16} \cdot 2 = 1.5625$

Compression ratio: $\frac{2}{1.5625} = 1.28$