EE4-08
EE9-SO14

IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING
EXAMINATIONS 2020

MSc and EEE/EIE PART IV: MEng and ACGI

DIGITAL IMAGE PROCESSING

Wednesday, 6 May 10:00 am

Time allowed: 3:00 hours

There are FOUR questions on this paper.

Answer THREE questions.

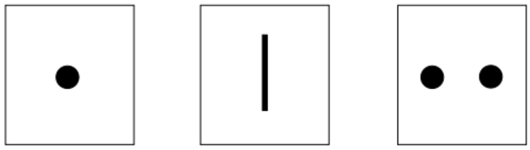
All questions carry equal marks

Any special instructions for invigilators and information for candidates are on page 1.

Examiners responsible	First Marker(s) :	P.T. Stathaki
	Second Marker(s) :	P.A. Naylor

1. (a) For an image given by the function $f(x,y) = (x+y)^3$ where x, y are continuous variables, evaluate $f(x,y)\delta(x-1, y-2)$ (2 points) and $f(x,y) * \delta(x-1, y-2)$ (2 points) where $\delta(x,y)$ is the two-dimensional (2D) Dirac Delta function. **[4]**

(b) Three binary images with value 255 on black areas and value 0 elsewhere are shown below. Describe roughly the form of the amplitude of the 2D DFT of these images by using properties of the DFT.



[3]

(c) Given the 2D impulse response $h(m,n)$

$$h(m,n) = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

(i) Calculate the Discrete Fourier Transform (DFT) of the given impulse response $h(m,n)$. **[4]**

(ii) Based on the result from (c)(i), determine if the given filter is a high-pass or a low-pass filter. Explain your reasoning. **[2]**

(d) 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). The three images are defined as follows:

$$f_1(x,y) = \begin{cases} r_1 & 1 \le x \le M, 1 \le y \le \frac{M}{2} \\ r_2 & 1 \le x \le M, \frac{M}{2} < y \le M \end{cases}$$

$$f_2(x,y) = r_3,\ 1 \le x \le M,\ 1 \le y \le M$$

$$f_3(x,y) = r_4,\ 1 \le x \le M,\ 1 \le y \le M$$

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) Find the images $g_1(x,y)$, $g_2(x,y)$ and $g_3(x,y)$ using the Karhunen-Loeve (KL) transform. Calculate all the steps of the transform in detail. **[5]**

(ii) Comment on whether you could obtain the result of (d)(i) above using intuition rather than by explicit calculation. **[2]**

2. (a) The following figures show: a $3-$bit image $f(x,y),\ x,y \in [0,4]$ of size 5×5, a Laplacian filter and a low-pass filter.

$$\begin{bmatrix} 3 & 7 & 6 & 2 & 0 \\ 2 & 4 & 6 & 1 & 1 \\ 4 & 7 & 2 & 5 & 4 \\ 3 & 0 & 6 & 2 & 1 \\ 5 & 7 & 5 & 1 & 2 \end{bmatrix}, \quad \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0.01 & 0.1 & 0.01 \\ 0.1 & 0.56 & 0.1 \\ 0.01 & 0.1 & 0.01 \end{bmatrix}$$

Compute the following:

(i) The output of a 3×3 mean (average) filter at $(2,2)$. **[1]**

(ii) The output of the 3×3 Laplacian filter shown above at $(2,2)$. **[1]**

(iii) The output of the 3×3 low-pass filter shown above at $(2,2)$. **[1]**

(b) Show that median filters are non-linear filters. **[2]**

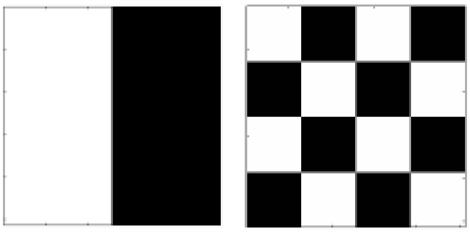
(c) Consider an image with 8 grey levels $r \in [0,7]$ with the histogram $(2,2,4,8,16,32,64,128)$.

(i) Apply histogram equalization to the given image. If the resulting intensities are not integers you can round them to the nearest integer. **[3]**

(ii) Using the result of (c)(i) above, explain the effect of histogram equalization on the given image. **[3]**

(d) Consider the two binary images of the same size shown below. The pixels shown with black have 0 intensity and the pixels shown with white have intensity equal to 1. The images shown are quite different, but their histograms are the same. Suppose that each image is blurred with a 3×3 averaging kernel. Would the histograms of the blurred images still be equal? Explain. If your answer is no, sketch the two histograms. In order to convolve the boundary pixels with the 3×3 kernel you can assume zero values outside the images.

[4]



(e) An image with normalized intensities in the range $[0,1]$ has the pdf of the form $p_r(r) = 2 - 2r,\ r \in [0,1]$. It is desired to transform the intensity levels of this image so that they will have a pre-specified pdf given as $p_z(z) = 3z^2,\ z \in [0,1]$.

(i) Assume continuous quantities and find the transformation (in terms of r and z) that will accomplish this requirement. **[3]**

(ii) Repeat part (e)(i) for the case when $p_z(z)$ is uniform. **[2]**

3. We are given the degraded version g of an image f such that

$$g = Hf + n$$

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. All images involved have size $N \times N$ after extension and zero-padding.

(a) (i) Describe how inverse filtering can be used to restore the degraded image above. **[4]**

(ii) Given knowledge of the exact degradation function, under what assumption can we perfectly restore the image? (2 points in total) How can we avoid erratic behaviour when the assumption is not met? (2 points in total) **[4]**

(iii) A particular image $f(x,y)$ is distorted by convolution with either the space invariant point function $h_1(x,y)$ or the function $h_2(x,y)$ where:

$h_1(x,y) = \delta(x,y) + \delta(x-1,y) + \delta(x+1,y) + \delta(x,y-1) + \delta(x,y+1)$

$h_2(x,y) = 5\delta(x,y) + \delta(x-1,y) + \delta(x+1,y) + \delta(x,y-1) + \delta(x,y+1)$

Assuming that the distorted images do not contain random additive noise then in one image the distortion can be effectively removed using an Inverse Filter, while in the other Pseudo Inverse Filter must be used. Which image must be filtered using Pseudo Inverse filter and why? **[4]**

(b) (i) Explain why when an image is corrupted by additive random noise, the noise is more visible in smooth regions with low contrast than in regions of high contrast and texture. **[2]**

(ii) Consider the Constrained Least Squares (CLS) filtering image restoration technique. Give the general expressions for both the CLS filter estimator and the restored image in both spatial and frequency domains and explain all symbols used. **[2]**

(iii) Explain the role of the regularization parameter a in Constrained Least Squares image restoration method. **[2]**

(iv) Propose a spatially adaptive Constrained Least Squares image restoration method which employs a spatially varying a in order to restore image g defined above. **[2]**

4. (a) A symbol source outputs the characters a to h depicted in Table 4.1 with their associated frequencies of occurrence based on the first 8 Fibonacci numbers.:

symbol	frequency
a	1
b	1
c	2
d	3
e	5
f	8
g	13
h	21

Table 4.1

(i) A Huffman code is used to represent the characters. Find the Huffman code demonstrating all the intermediate steps. For the construction of Huffman code assign 1 to the joint symbol when you merge symbols. Furthermore, when you merge symbols a and b assign 1 to symbol a. **[3]**

(ii) What is the sequence of characters corresponding to the code 110111100111010? **[3]**

(iii) Determine the redundancy of the Huffman code for this example. Comment on the efficiency of Huffman code for this particular set of symbols. **[3]**

(b) A compression method is required in order to encode a text message before transmitting it over a network. The message consists of the characters depicted in Table 4.2 with their associated frequencies of occurrence.

symbol	frequency
a	5
b	9
c	12
d	13
e	16
f	45

Table 4.2

(i) A Huffman code is used to represent the characters. Find the Huffman code demonstrating all the intermediate steps. **[3]**

(ii) Determine the redundancy of the Huffman code for this example. Comment on the efficiency of Huffman code for this particular set of symbols. **[3]**

(iii) If we assume that each character in the transmitted text message without using any coding scheme, requires one byte, calculate the number of bits that we will save if we apply Huffman coding. **[5]**