APPM4058A & COMS7238A - Digital Image Processing
Class Test I

April 26, 2019

Student Number:

Instructions

- Specified course materials, i.e., lecture slides are allowed to be used during the test. When these materials are not in print, students are allowed to view them using computers.
- Electronic devices other then those specified above, and Internet are not allowed during the test.
- You may use a calculator.
- Test duration: two hours.
- The total mark for this test is 60, and bears a weight of 20% towards your final mark.

Total Mark: [ ]/**60**

Part I [32]

1. For Image A shown in Table 1, find a transformation that will change its histogram to match the one shown in Table 2. Give the transformed image as 5×5 image. Assume that the processed image can only take integer values between 0 and 3 (including 3), give the histograms of the original and processed images, respectively. [10]

0	0	1	1	2
0	1	1	2	2
1	1	2	3	1
1	2	3	1	1
2	3	1	0	0

Table 1: Image A

Gray level f	0	1	2	3
Histogram $h(f)$	$15/25$	0	0	$10/25$

Table 2: Desired histogram

Answer:

(a) The process of deriving the transformation is shown in the Table 3, the transformation function is columns 0 and 7. (Each value in column 7 is 1 mark, 4 marks in total.)

Table 3: The histogram matching process

0 r_k	1 Original $p_r(r_k)$	2 CDF $T(r_k)$	3 CDF$*$(L-1) s_k	4 Desired $p_z(z_k)$	5 CDF $G(z_k)$	6 CDF$*$(L-1) v_k	7 Map(z_k)	8 Actual
0	$5/25$	$5/25$	1	$15/25$	$15/25$	2	0	$16/25$
1	$11/25$	$16/25$	2	0	$15/25$	2	0	0
2	$6/25$	$22/25$	3	0	$15/25$	2	3	0
3	$3/25$	1	3	$10/25$	1	3	3	$9/25$

(b) See Table 4. (On completing the previous question, 2 marks for a correct answer, 1 mark for incorrect answer. 0 mark if the previous question is not completed.)

Table 4: The output image

0	0	0	0	3
0	0	0	3	3
0	0	3	3	0
0	3	3	0	0
3	3	0	0	0

(c) See columns 2 (1 mark) and 8 (2 marks) in Table 3 for the original histogram and the one after performing histogram matching, respectively.

2. Filter $h = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$ can be separated into two 1D filters. That is $h = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$. The following array represents a small 4-bit gray level image.

$$I = \begin{array}{ccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

Perform spatial filtering on Image I using the filter h. For this, you are requested to do the filtering using the two 1D filters instead of the 2D filter. Show the results after applying each of the 1D filters. [6]

Answer:

(a) Compute the 1D convolution using the row filter $\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$: we need to rotate the filter by 180° first. However, the row filter is symmetric. In order to move the filter over I, we need to make sure at least one of the elements in the filter overlaps with I. Thus, we pad I with 0s along the horizontal direction. The filter is moved row by row along the horizontal direction.

$$I = \begin{array}{ccccccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \rightarrow \begin{array}{ccccccccccc} 0(1) & 0(2) & 0(1) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$\begin{array}{ccccccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0(1) & 0(2) & 1(1) & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$\begin{array}{ccccccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 4 & 8 & 8 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \rightarrow \begin{array}{ccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 4 & 8 & 8 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

(b) Next, we perform convolution using the column vector $\begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$ column by column. We rotate the filter by 180°, which becomes $\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$. We also need to pad the image along the vertical direction:

$$\begin{array}{ccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0(-1) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0(0) & 0 & 0 & 0 & 0 & 0 \\ 0 & 1(1) & 4 & 8 & 8 & 3(-1) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0(0) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0(1) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \rightarrow \begin{array}{ccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 4 & 8 & 8 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -4 & -8 & -8 & -3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

3. Suppose that a digital image is subjected to histogram equalization. Show that a second pass of histogram equalization will produce exactly the same result as the first pass. [6]

Answer: Let n be the total number of pixels and let n_{r_j} be the number of pixels in the input image with intensity r_j. Then, the histogram equalization transformation is

$$s_k = T(r_k) = \sum_{j=0}^{k} n_{r_j}/n = \frac{1}{n}\sum_{j=0}^{k} n_{r_j}.$$

Since every pixel (and not others) with value r_k is mapped to value s_k, it follows that $n_{s_k} = n_{r_k}$. A second pass of histogram equalization would produce values v_k according to the transformation

$$v_k = T(s_k) = \frac{1}{n}\sum_{j=0}^{k} n_{s_j}.$$

But $n_{s_j} = n_{r_j}$, so

$$v_k = T(s_k) = \frac{1}{n}\sum_{j=0}^{k} n_{r_j} = s_k,$$

which shows that a second pass of histogram equalization would yield the same result.

4. (a) Discuss the process of filtering in the frequency domain. [3]

Answer: Given an image I, the process of filtering in frequency domain involves the following:

i. Preprocessing – padding the image;
ii. Compute the DFT of I, denoted by F;
iii. Obtain a frequency domain filter, H;
iv. Compute $G = F * H$, element-wise multiplication (convolution theorem);

v. Compute the inverse DFT of G.

vi. Post-processing – Crop the output of the inverse transform to the size of the image I.

(b) What are the differences between filtering in the frequency domain and filtering in the spatial domain? [2]

Answer: The main difference is filtering in spatial domain is a local operation, and filtering the frequency domain is a global operation.

5. When filtering in the frequency domain, it is preferred to do image padding.

(a) Why there is a need to do image padding when filtering in the frequency domain? [3]

Answer: DFT assumes the input functions to be periodic, where the period is equal to the length of the functions. Convolving periodic functions can cause interference between adjacent periods if the periods are close with respect to the duration of the nonzero parts of the functions. This can be visualized as when the nonzero part of one period of the filter moves past the nonzero part of the function, the nonzero part of the adjacent period from the filter already enters the nonzero part of the function. This can cause interference (wraparound error). The interference can be avoided by padding the functions with zeros.

(b) Describe how image padding is done. [2]

Answer: Assume the input function $f(x,y)$ is of size $A \times B$, and the filter $h(x,y)$ is of size $C \times D$. We form two padded functions, both of size $P \times Q$, where $P \geq A+C-1$ and $Q \geq B+D-1$. Padded image for f is usually done by padding zeros at the ends of rows and columns, i.e., the original image will be fit in the top left corner of the padded image. For the filter in spatial domain, we first pad the filter with zeros as described above, and then compute the DFT to obtain the corresponding frequency domain filter. However, for the filters directly obtained in the frequency domain, it is of the same size as the padded image. So there is no padding in the spatial domain involved. As a result, in this case, it is not guaranteed that wraparound error is eliminated completely.

Part II [12]

A set of 3×3 filters (their origins are all at the center) are given in Table 5.

0	0	0
0	1	0
0	0	0

J

0	1	0
1	0	1
0	1	0

K

Table 5: Filters J and K.

1. Suppose B is a binary image and J and K are kernels specified as in Table 5. Explain the purpose of operation $B \ominus J \cap B^c \ominus K$ where B^c is the complement of B. [3]

Answer: The operation is hit-or-miss transform (1 mark). Its purpose is to detect isolated white pixels with intensity value 1 (2 marks).

2. Apply Mask B in Table 7 to binary image I in Table 6 to implement dilation operation. Put your answer in Table 8. [5]

0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0
0	0	1	0	1	0	0	0
0	0	0	0	1	0	0	0
0	0	1	1	1	0	0	0
0	0	1	0	1	0	0	0
0	0	1	1	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Table 6: Image I

0	1	0
1	1	0
0	0	1

Table 7: Mask B

Table 8: Image I after dilation

Answer: The dilation result is shown in Table 9.

0	0	0	0	1	0	0	0
0	0	1	1	1	0	0	0
0	1	1	1	1	1	0	0
0	0	1	1	1	1	0	0
0	1	1	1	1	1	0	0
0	1	1	1	1	1	0	0
0	1	1	1	1	1	0	0
0	0	0	1	1	1	0	0
0	0	0	0	0	0	0	0

Table 9: Image I_d

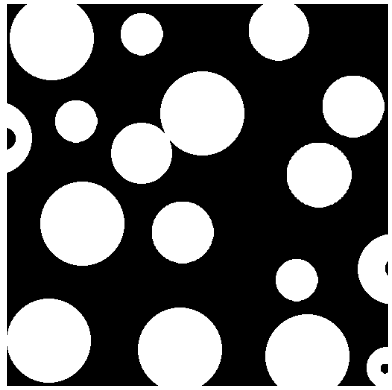

Figure 1: Circles

3. What is the functionality of each filter given below? Explain your reasoning. [4]

$$B_1 = \frac{1}{4}\begin{bmatrix} -1 & -2 & -1 \\ -2 & 16 & -2 \\ -1 & -2 & -1 \end{bmatrix} \quad B_2 = \begin{bmatrix} -1 & -3 & -1 \\ 0 & 0 & 0 \\ 1 & 3 & 1 \end{bmatrix}$$

Answer: B_1 is a smoothing filter (1 mark), because the coefficients sum to 1 (1 mark). B_2 is a edge detection filter (1 mark), because the coefficients sum to 0 (1 mark), it horizontally smoothing, vertically edge detecting to detect horizontal edges.

Part III [16]

1. Using the definition of binary dilation to prove that $A \oplus (B \oplus C) = (A \oplus B) \oplus C$. [6]

   Answer: Let $x \in A \oplus (B \oplus C)$ if and only if there exists $a \in A$, $b \in B$, and $c \in C$ such that $x = a + (b + c)$ (2 marks). $x \in (A \oplus B) \oplus C$ if and only if there exists $a \in A$, $b \in B$, and $c \in C$ such that $x = (a + b) + c$ (2 marks). But $a + (b + c) = (a + b) + c$ since addition is associative (2 marks). Therefore, $A \oplus (B \oplus C) = (A \oplus B) \oplus C$.

2. Are the following two expressions True or False?

   (a) $A \subseteq A \circ B$ [1]

   Answer: False.

   (b) $(A \oplus B) \ominus C \subseteq A \oplus (B \ominus C)$ [1]

   Answer: False.

3. Figure 1 shows Image ‘Circles’. In this image, there are a number of incomplete circles around the border of the image. Based on morphological operations, design an algorithm to clear these incomplete circles so that the resulting image will contain only those complete circles. [8]