VASAVI COLLEGE OF ENGINEERING (AUTONOMOUS)

IBRAHIMBAGH, HYDERABAD-31

Department of Computer Science and Engineering

#### Name of the Course: Image Processing

Assignment – 3

Name of the Faculty: C.Gireesh Date of submission: 29-04-2024

Class: B.E CSE-A VI Sem Time: 4.20 PM

Academic Year: 2023-24

**Set-1(**Top 10 Students: **1602-21-733-005, 012, 13, 20, 26, 29, 32, 36, 37, 63)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1. | One variation of run-length coding involves (1) coding only the runs of 0’s or 1’s (not both) and (2) assigning a special code to the start of each line to reduce the effect of transmission errors. One possible code pair is (*x* ,*r* ), *k k* where *xk* and *rk* represent the *k*th run’s starting coordinate and run length, respectively. The code (0, 0) is used to signal each new line.  **(a)** Derive a general expression for the maximum average runs per scan line required to guarantee data compression when run-length coding a 2*n* × 2*n* binary image.  **(b)** Compute the maximum allowable value for *n* = 10. | 2 | 3 | CO4 |
| 2. | During acquisition, an image undergoes uniform linear motion in the vertical direction for a time *T*1. The direction of motion then switches  to the horizontal direction for a time interval *T*2 . Assuming that the time it takes the image to change directions is negligible, and that shutter opening and closing times are negligible also, give an expression for the blurring function, *H*(u,v). | 3 | 3 | CO5 |

**Set-2 (1602-21-733-001,02,03,04,06,07,08)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1. | Do the following:  **(a)** Construct the entire 4-bit Gray code.  **(b)** Create a general procedure for converting a Gray-coded number to its binary equivalent and use it to decode 0111010100111. | 2 | 3 | CO4 |
| 2. | Apply Least square filtering for the given corrupted image and the degradation function . | 3 | 3 | CO5 |

**Set-3 (1602-21-733-009, 10,11,14,15,16,17)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1. | Consider the simple 4 × 8, 8-bit image:  21 21 21 95 169 243 243 243  21 21 21 95 169 243 243 243  21 21 21 95 169 243 243 243  21 21 21 95 169 243 243 243  **(a)** Compress the image using Huffman coding. | 2 | 3 | CO4 |
| 2. | Apply Inverse filtering for the given corrupted image and the degradation function to obtain the restored image . | 3 | 3 | CO5 |

**Set-4(18,19,21,22,23,24,25)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1 | Consider the simple 4X8, 8-bit image:  21 21 21 95 169 243 243 243  21 21 21 95 169 243 243 243  21 21 21 95 169 243 243 243  21 21 21 95 169 243 243 243  Compute the entropy of the image. | 2 | 3 | CO4 |
| 2 | Apply Weiner filter for the given corrupted image and the degradation function to obtain the restored image . | 3 | 3 | CO5 |

**Set-5(27,28,30,31,33,34,35)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1 | Using the Huffman codes given in the table, decode the encoded string 0101000001010111110100   |  |  | | --- | --- | | **symbol** | **code** | | a1 | 011 | | a2 | 1 | | a3 | 01010 | | a4 | 0100 | | a5 | 01011 | | a6 | 00 | | 2 | 3 | CO4 |
| 2 | Given the corrupted image . Apply the midpoint filter to obtain the restored image | 3 | 3 | CO5 |

**Set-6(38,39,40,41,42,43,44)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1 | Compute exponential Golomb code | 2 | 3 | CO4 |
| 2 | Given the degraded image . Apply the arithmetic mean filter to obtain the restored image | 3 | 3 | CO5 |

**Set-7(46,47,48,49,51,52,53)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1 | Decode the message 0.068 given the coding model   |  |  | | --- | --- | | **Source symbol** | **Probability** | | a1 | 0.2 | | a2 | 0.2 | | a3 | 0.4 | | a4 | 0.2 | | 2 | 3 | CO4 |
| 2 | Apply Weiner filter for the given corrupted image and the degradation function to obtain the restored image . | 3 | 3 | CO5 |

**Set-8(54,55,56,57,58,59,60)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1 | Decode the message 0.23355 given the coding model   |  |  | | --- | --- | | **Source symbol** | **Probability** | | A | 0.2 | | E | 0.3 | | I | 0.1 | | O | 0.2 | | U | 0.1 | | ! | 0.1 | | 2 | 3 | CO4 |
| 2 | Consider the RGB cube in Fig. 6.8 and answer each of the following questions.  **(a) \*** Describe how the gray levels vary in each of the R, G, and B primary images that make up the front face of the color cube (this is the face closer to you). Assume that each component image is an 8-bit image.  **(b)** Suppose that we replace every color in the RGB cube by its CMY color. This new cube is displayed on an RGB monitor. Label with a color name the eight vertices of the new cube that you would see on the screen. | 3 | 3 | CO5 |

**Set-9(61, 62,64,65,66,67,135)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1 | Compute compression ratio and relative redundancy of an image.   |  |  |  |  | | --- | --- | --- | --- | | intensity | probability | Code1 | Code2 | | 87 | 0.25 | 01010111 | 01 | | 128 | 0.47 | 10000000 | 1 | | 186 | 0.25 | 11000100 | 000 | | 255 | 0.03 | 11111111 | 001 | | 2 | 3 | CO4 |
| 2 | Apply Inverse filtering for the given corrupted image and the degradation function to obtain the restored image . | 3 | 3 | CO5 |

**Set-10(136,301,302,303,304,305,306,307)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q. No | Questions | Marks | BTL | Course Outcome |
| 1 | Use the LZW coding algorithm to encode the 7-bit ASCII string “aaaaabaaaaa” | 2 | 3 | CO4 |
| 2 | Apply Least square filtering for the given corrupted image and the degradation function . | 3 | 3 | CO5 |