|  |
| --- |
| **Digital Image Processing (CS4055)** |
| **Course Instructor(s):** |
| Mr. Usama Imtiaz  **Section(s): AI-22 All Sections** |

|  |  |
| --- | --- |
| **Sessional-II Exam** | |
| **Total Time (Hrs):** | **1** |
| **Total Marks:** | **55** |
| **Total Questions:**  **Date:** Apr 8, 2025 | **7** |

**\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Roll No Course Section Student Signature**

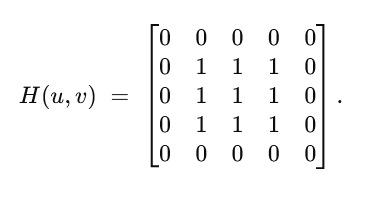
**Do not write below this line.**

**Attempt all the questions.**

**Q1:** You have a **5×5 discrete Fourier spectrum** F(u,v) (magnitude only shown here). The coordinates (u,v) range from −2…2 in both directions. The table below shows the magnitude values **[10+4 marks]**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **(u,v)** | **-2** | **-1** | **0** | **1** | **2** |
| **-2** | 2 | 3 | 7 | 3 | 2 |
| **-1** | 3 | 10 | 20 | 10 | 3 |
| **0** | 7 | 20 | 50 | 20 | 7 |
| **1** | 3 | 10 | 20 | 10 | 3 |
| **2** | 2 | 3 | 7 | 3 | 2 |

1. Apply an Ideal Low-Pass Filter with a cutoff radius D0=1.5. Zero out any frequency whose distance is greater or equal than 1.5



1. In the table below, mark which frequencies remain and which are set to 0. You only need to fill “KEEP” or “ZERO” in the corresponding cell.

A screenshot of a calculator

AI-generated content may be incorrect.

1. After applying this filter, you inverse-transform back to the spatial domain. In a 2–3 sentence (max) calculation note (not an explanation essay), describe how the ringing effect can appear in the output image.

 **Inverse DFT** of this sharply truncated frequency spectrum typically introduces **ringing** in the spatial domain, because an **Ideal (abrupt) cutoff** in frequency corresponds to a **sinc-like kernel** in the spatial domain with oscillations (overshoot/undershoot).

 This phenomenon is the “ringing effect.”

**Q2:** Noise-Only Degradation & Spatial Filters **[6+5 marks]**

You captured a **3×3** patch of an image affected by Gaussian noise with mean 0 and variance σ2. The patch is:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Col 1** | **Col 2** | **Col 3** |
| **Row 1** | 95 | 102 | 98 |
| **Row 2** | 100 | 105 | 250 |
| **Row 3** | 103 | 97 | 101 |

1. Apply each of the following filters to the center pixel (Row 2, Col 2) only (ignore boundary for simplicity):
   * Harmonic Mean Filter (3×3)
   * Contra-Harmonic Mean Filter (Q = 1.5)

A math equation with numbers and lines

AI-generated content may be incorrect.

A math equations and formulas

AI-generated content may be incorrect.

A math equations and formulas

AI-generated content may be incorrect.

1. List the resulting single output value (rounded to nearest integer) for each filter in the table below:

|  |  |
| --- | --- |
| **Filter** | **Center Output** |
| Harmonic Mean (3×3) | 108.78 or 108 |
| Contra-Harmonic Mean (Q=1.5) | 119.16 or 119 |

1. State a direct, one-line remark on the effect of each filter with respect to: Smoothing of Gaussian noise and handling potential outliers or very large pixel values (like the “250” in the table)

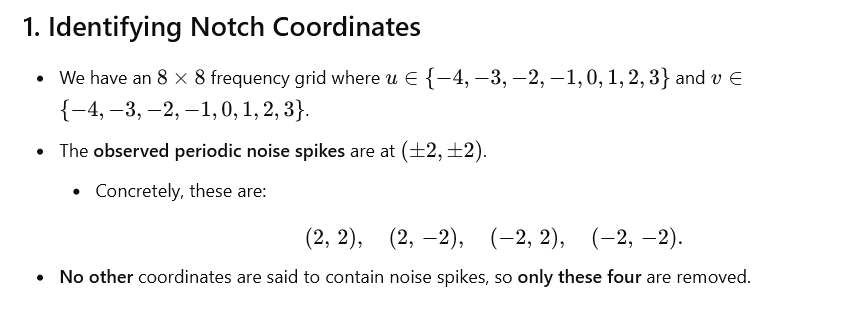
* **Harmonic Mean**:
  + Noise Smoothing: Effective at suppressing large outlier (salt) noise, as large values contribute very small fractions (1/xi).
  + Outlier Handling: The “250” pixel is down-weighted in the harmonic mean, so the result is noticeably lower.
* **Contra-Harmonic (Q = 1.5)**:
  + Noise Smoothing: With Q>1Q>1, large values are **emphasized**. This filter targets pepper noise more effectively.
  + Outlier Handling: A single large pixel (250) strongly increases the numerator (xi2.5), yielding a higher output than other means.

**(Example wording: “Arithmetic Mean Filter spreads the influence of the 250 pixel across the result more strongly than a harmonic filter would,” etc.)**

**Q3:** An image’s Fourier spectrum (magnitude) shows periodic noise spikes at frequencies (u,v)=(±2 , ±2). You want to remove them with notch rejects. The size of the spectrum is 8×8 with coordinates −4…+3−4…+3. **[05 marks]**

Identify which coordinates in the 8×8 frequency plane must be “notched out” (set to zero).

1. Provide the final coordinate list.



After notching, you inverse DFT and get the restored image.

1. Suppose the total power from those spikes was 80. If the original image total power was 800, compute the new total power in the restored image.

A white paper with black text

AI-generated content may be incorrect.

**Q4:**. You are given an image of size M x N. You applied Low pass filtering to the image in frequency

domain using a Gaussian Low pass filter. The cutoff frequency was D0. Next, you applied the same Gaussian Low pass filter repeatedly to the image. This means that you apply filter in frequency domain and then go back to spatial domain, then come back again in frequency domain, apply the filter and then go back and so on. Suppose you applied the filter “K” number of times. You can assume that K is a large number. Can you guess what will happen to the image?  **[05 marks]**

After **repeatedly applying** the same Gaussian low-pass filter in the frequency domain (and transforming back to spatial between each pass), the image will become **increasingly blurred**. As K→∞K\to\infty, it effectively converges to a **nearly constant (flat) intensity** image in which all fine details (higher frequencies) have been lost.

**Q5:** Consider a pair of input (Left image) and output (right image) image of a certain procedure.

Which operations are most likely involved in this procedure? Why? (Max 2-3 lines) **[05 marks]**

****

A multi‐step procedure is **noise smoothing** (e.g., median filtering), **edge detection** (e.g., Sobel, Laplace etc), and **thresholding**. The smoothing step suppresses noise, the edge detector highlights contours, and thresholding produces a cleaner, high‐contrast edge map.

**Q6:** The following image is being distorted by a noise. **[05 marks]**

|  |  |
| --- | --- |
| 1. Write down the name of the noise. 2. Clearly write down the steps for removing this noise(also explain which filter will be used and why) |  |

1. The visible diagonal pattern is a form of **periodic noise** (often seen as stripes or repetitive interference in the spatial domain).
2. Steps to Remove This Noise
3. **Transform to Frequency Domain**
   * Compute the 2D DFT (Fourier transform) of the image.
   * Periodic stripe noise appears as distinct spikes away from the center of the spectrum.
4. **Identify and Notch Out the Noise Frequencies**
   * Locate the spikes corresponding to the diagonal pattern.
   * Apply **Notch Filters** (or a narrow Band‐Reject Filter) at those specific frequency coordinates. This zeroes out (or suppresses) the unwanted periodic components.
5. **Inverse Transform**
   * Perform the inverse DFT to return to the spatial domain.
   * The resulting image will have most of the diagonal stripes (periodic noise) removed.

**Why a Notch Filter?**

* A **Notch Filter** precisely targets and attenuates (or removes) the narrow set of frequencies that cause the diagonal stripes, while preserving the rest of the image’s frequency content. This is typically more effective than broad smoothing or other methods, which may degrade overall image detail.

**Q7: [5+5 marks]**

|  |  |
| --- | --- |
|  |  |
| Multiplication of the disc (black: 0, white: 1) with an image’s DFT in frequency domain achieves:  Because the central (low‐frequency) region is black (i.e., multiplied by zero) and the rest is white (multiplied by one), this effectively **removes low frequencies and preserves high frequencies**. In other words, it acts as an **Ideal High‐Pass Filter** in the frequency domain. | Multiplication of the mask below (black: 0, white: 1) with an image’s DFT in frequency domain achieves:  Because only the four white “spots” are passed (multiplied by 1) and all other frequencies are zeroed out, this **selects just those few discrete frequency components** in the DFT. The inverse‐transformed result will be a superposition of a small number of sinusoids corresponding to the white spots’ locations, effectively acting as a **multi‐point “notch‐pass”** that removes all other frequency information. |