CSE-4875 || Pattern Recognition and Image Processing || Mid Term Notes *(Prev. Sol also)*

Prepared By- **Sorowar Mahabub,** *C201032 & Others*

Spring-23

1(a): 'One picture is worth more than ten thousand words'. Explain the fields of image processing used in the statement. [Marks- 2]

The statement emphasizes the power of visual information over verbal communication. In the context of image processing, it underscores the importance of visual data and the impact it can have compared to textual descriptions. It means you can tell an entire story with just one picture and impart a concept or idea much more effectively. **In the context of image processing, relevant fields include:**

Image Compression: Efficiently representing visual information for storage or transmission.

Image Enhancement: Improving the quality or appearance of images for better visual communication.

Image Recognition: Identifying objects or patterns within images for automated understanding.

Image Analysis: Extracting meaningful information from images, aiding in decision-making.

Image Segmentation: Dividing images into distinct regions for focused analysis.

Visual Perception Models: Understanding how humans perceive and interpret visual information.

1(b): Find out the resolution of a 20'' monitor working on 1024 x 768. [Marks- 2]

The formula to calculate resolution is: Resolution = sqrt (horizontal_resolution^2 + vertical_resolution^2) / diagonal size

Using this formula, we can calculate the resolution of a 20-inch monitor with a resolution of1024 x 768 as follows:

Resolution = sqrt ($1024^2 + 768^2$) / 20 PPI = 1280/20 PPI = 64 PPI

Therefore, the resolution of the 20-inch monitor with a resolution of 1024 x 768 is 64 PPI.

1(c): Explain the mathematical model of an analog image. How can we convert an analog image to a digital image? [Marks- 3] Au'22 & sp'23

Mathematical model of an analog image:

- Illumination & reflectance components:
  - Illumination: i(x,y)
  - Reflectance: r(x,y)
  - f(x,y) = i(x,y) · r(x,y)
  - $0 < i(x,y) < \infty$
    and $0 < r(x,y) < 1$
    (from total absorption to total reflectance)

- Amount of light incident by object or point:

Illumination (meaning:source of light) i(x,y)
The value of i(x,y) is between 0 and α.
Some typical values of i(x,y)

Clear Sunny day	10,000	ft-candles.
Cloudy day	1,000	ft-candles.
Clear full moon	0.01	ft-candles.
Typical office	100	ft-candles.

- Amount of light reflected on object or point:

Reflectance or r(x,y). and 0<r(x,y)<1.
r=0 no reflection; total absorption
r=1 full reflection; no absorption
Some typical values of r(x,y)

Black velvet	0.01
Stainless steel	0.65
White wall	0.8
Silver-plated metal	0.9
Snow	0.93

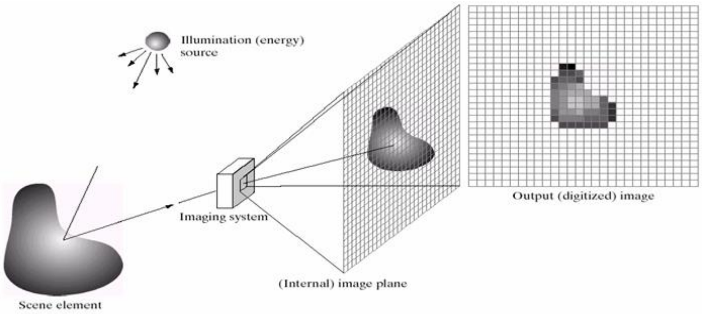
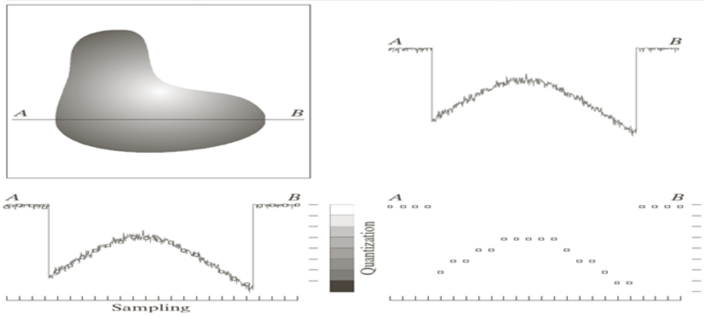


FIGURE 2.15 An example of the digital image acquisition process. (a) Energy ("illumination") source. (b) An element of a scene. (c) Imaging system. (d) Projection of the scene onto the image plane. (e) Digitized image.



Conversion of Analog Image into Digital Image:

There are numerous ways to acquire images but our objective to generate digital images from sensed data. To create digital image, we need to convert the continuous sensed data into digital form. **This involves two processes: sampling and quantization.** The basic idea behind sampling and quantization is illustrated in ***above Fig. 2.16***. Fig. 2.16 (a) shows a continuous image, f(x, y), that we want to convert digital form. An image may be continuous with respect to the x- and y- coordinates and also in amplitude. To convert it to digital form, we have to sample the function in both coordinates and in amplitude.

Digitizing the coordinate values is called sampling. || Digitizing the amplitude values is called the quantization.

To understand sampling and quantization you can check out the given link: https://www.youtube.com/watch?v=KWc9SOOLfLw

1(d): "On Aug. 23, 2023, the Indian Space Research Organization's (ISRO) Chandrayaan- 3 3 mission landed near the Moon's south pole, and a day later, the rover took a walk on the lunar surface." - Write the applications of image processing used in this expedition. [Marks- 3]

The applications of image processing used in the Chandrayaan-3 mission to the Moon's south pole include:

Terrain Mapping: Processing images to create detailed maps of the lunar surface, aiding in navigation and exploration.

Obstacle Detection: Analyzing images to identify potential obstacles or hazards on the lunar terrain, ensuring safe navigation for the rover.

Geological Analysis: Processing images to analyze the composition and structure of lunar rocks and formations, providing valuable scientific insights.

Navigation and Localization: Using image processing techniques for precise localization of the rover on the lunar surface, facilitating accurate navigation.

Panoramic Imaging: Stitching together multiple images to create panoramic views of the lunar landscape, enhancing the understanding of the mission's surroundings.

Image Compression: Efficiently transmitting and storing large volumes of image data captured by the rover, optimizing communication bandwidth and storage resources.

Communication Enhancement: Image processing techniques may have been used to improve the quality of images transmitted back to Earth, ensuring clearer and more informative visuals.

These applications of image processing played crucial roles in enabling the success of the Chandrayaan-3 mission by providing essential visual information for navigation, exploration, and scientific analysis on the lunar surface.

OR, Write the similarity and difference between eyes and camera. [Marks- 3] Au'22 & sp'23

Similarity between eyes and camera:

Eyes	Camera
1.The lens and cornea focus light onto the back of the retina. 2.Muscles change the shape of the lens to focus the image. 3.The mussels in the iris regulate the amount of light that reaches the retina.	1.The lens focuses light onto a light-sensitive surface called film 2.The lens of a camera can be moved back or forward to focus images 3.The aperture, like the iris, regulate the amount of light that reaches the retina

Difference between eyes and camera:

Main difference: An eye is a physiological computational element but a camera is a mechanical device. Eye doesn't work independently and its operation correlated with brain, whereas a camera that does what let it to do

Retina and film: Retina can automatically change its sensitivity to light sensitivity depending upon the amount of illumination presented to it. On the other hand, in the camera can never change their sensitivity automatically

Binocular vision: an eye has a blind spot through which all nerves leave to brain. So, normally all living beings have two eyes. The part of image that blind spot of one eye cannot be covered by the second eye. Unlike eyes, cameras don't have any blind spot and hence need only one lens. That is most cameras don't have "binocular vision" which is a key characteristic of eye.

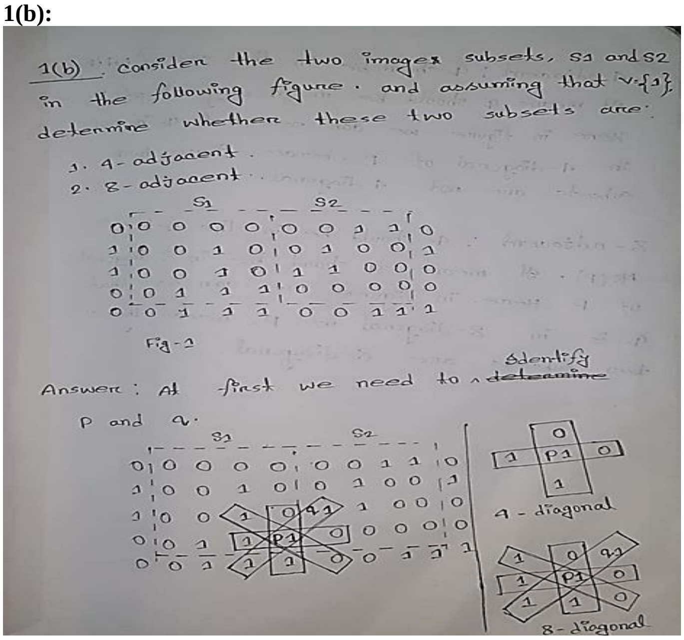
Automatic focusing: Eye has automatic focusing capability which is controlled by brain. It is neurological. Eyes ca n focus objects at any distance automatically but in the case of cameras it is to be accomplished by a photographer. So, it is manual in cameras.

Autumn-22:

1(a): Image processing is developed for improvement of pictorial information for human interpretation" – explain.

Image processing involves techniques and algorithms designed to enhance pictorial information to make it more visually appealing and easier for humans to interpret. This can include improving image quality, enhancing details, reducing noise, adjusting brightness and contrast, and extracting useful information from images for better understanding and analysis. Overall, the goal of image processing is to optimize visual information to aid human perception and interpretation.

1(b):

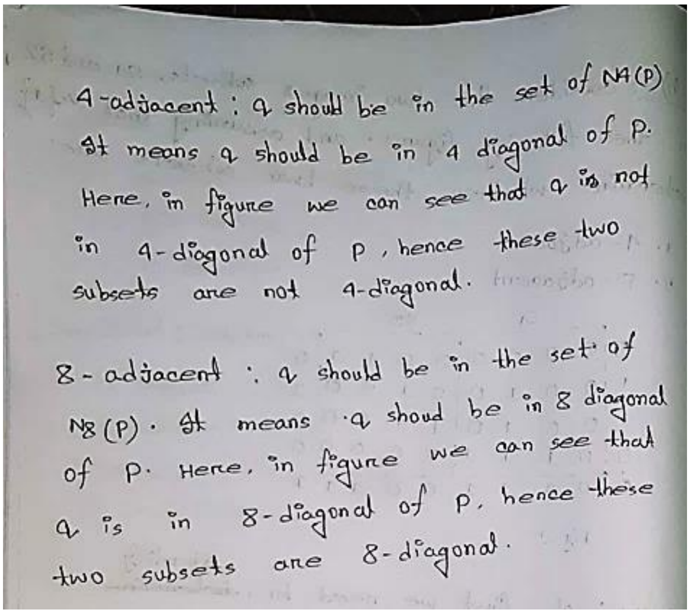

1(b) Consider the two images subsets, S1 and S2 in the following figure. and assuming that V={1}, determine whether these two subsets are:

1. 4-adjacent.
2. 8-adjacent.

S1					S2				
0	0	0	0	0	0	0	1	1	0
1	0	0	1	0	0	1	0	0	1
1	0	0	1	0	1	1	0	0	0
0	0	1	1	1	0	0	0	0	0
0	0	1	1	1	0	0	1	1	1

Fig-1

Answer: At first we need to identify ~~determine~~ p and q.


4-adjacent: q should be in the set of N4(p). It means q should be in 4 diagonal of p. Here, in figure we can see that q is not in 4-diagonal of p, hence these two subsets are not 4-diagonal.

8-adjacent: q should be in the set of N8(p). It means q should be in 8 diagonal of p. Here, in figure we can see that q is in 8-diagonal of p, hence these two subsets are 8-diagonal.

1(c): Briefly state the working principle of camera.

The working principle of a camera involves capturing and recording light patterns from a scene onto a photosensitive medium. This is typically done through a series of steps:

Light enters the camera through the lens: The lens focuses light onto the photosensitive medium, such as a digital sensor or film.

Image formation: Light rays passing through the lens converge to form an image on the medium.

Exposure control: The camera controls the amount of light reaching the medium by adjusting the size of the aperture (opening) and the duration of exposure.

Photosensitive medium response: The medium reacts to light exposure, capturing the intensity and color information of the scene.

Image processing (for digital cameras): The captured light information is converted into digital data, processed, and stored as an image file.

Output: The processed image can then be displayed on a screen or printed.

Overall, the camera's working principle involves capturing and recording light patterns to create images of the scene being photographed.

1(d): A common measure of transmission for digital data is the baud rate, defined as symbols (bits in our case) per second. As a minimum, transmission is accomplished in packets consisting of a start bit, a byte (8 bits) of information, and a stop bit. Using these facts, answer the following:
How many seconds would it take to transmit a sequence of 500 images of size 1024 × 1024 pixels with 256 intensity levels using a 3M-baud (106 bits/sec) baud modem? (This is a representative medium speed for a DSL. (Digital Subscriber Line) residential line.

To calculate the time required to transmit a sequence of 500 images, we need to determine the total number of bits in the sequence and then divide by the baud rate.

First, let's calculate the number of bits in one image:
Each pixel has 8 bits (256 intensity levels).

Since we're using a start bit, 8 data bits, and a stop bit for each byte, the total number of bits per image becomes:
1 start bit + 8 data bits + 1 stop bit = 10 bits per byte.

Image size: 1024 x 1024 pixels.

Total bits per image = 1024 pixels * 1024 pixels * 10 bits/pixel
= 10,485,760 bits

Now, to transmit 500 images:

Total bits for 500 images = 500 * 10485760 bits = 5,242,880,000 bits.

Finally, we can calculate the time required to transmit these bits using the baud rate:

Time = Total bits / Baud rate

Time = 5,242,880,000 bits / (3 * 10^6 bits/sec)

≈ 1747.6267 seconds

So, it would take approximately 1747.6267 seconds to transmit a sequence of 500 images with the given specifications using a 3M-baud modem**.**

OR, if a color image has 2160 x 3240 pixels with resolution 200 dpi. What will be the space taken by the image? What will be the size of the image?

To calculate the space taken by the image and its size, we need to consider the resolution and color depth. Given:

- Image dimensions: 2160 x 3240 pixels || - Resolution: 200 dots per inch (dpi)

1. Space taken by the image:

To calculate the space taken by the image, we first need to find out the total number of pixels in the image:

Total pixels = Width x Height

Total pixels = 2160 pixels x 3240 pixels = 6,998,400 pixels

Since the image is in color, we also need to consider the color depth. A common color depth for images is 24 bits per pixel (8 bits per channel for Red, Green, and Blue).

Total bits = Total pixels x Color depth

Total bits = 6,998,400 pixels x 24 bits/pixel = 167,961,600 bits

Converting bits to bytes (1 byte = 8 bits):

Total bytes = Total bits / 8

Total bytes = 167,961,600 bits / 8 bits/byte = 20,995,200 bytes

Converting bytes to megabytes (1 megabyte = 1024 kilobytes, 1 kilobyte = 1024 bytes):

Total megabytes = Total bytes / (1024 * 1024)

Total megabytes = 20,995,200 bytes / (1024 * 1024) ≈ 20 MB (rounded to nearest integer)

Therefore, the space taken by the image is approximately 20 megabytes.

2. Size of the image:

The size of the image in terms of dimensions (width x height) remains the same, which is 2160 x 3240 pixels.

However, if you're referring to the physical size of the image when printed at the specified resolution (200 dpi), we can calculate it as follows:

Image width in inches = Image width in pixels / Resolution (dpi)

Image width in inches = 2160 pixels / 200 dpi ≈ 10.8 inches

Image height in inches = Image height in pixels / Resolution (dpi)

Image height in inches = 3240 pixels / 200 dpi ≈ 16.2 inches

Therefore, the size of the image when printed at 200 dpi would be approximately 10.8 inches x 16.2 inches.

Sp'23

2(a): Justify the statement, "Applying Low-pass filter on an image result in a blurrier Image". Explain with proper example.

The statement "Applying a low-pass filter on an image result in a blurrier image" can be justified based on the fundamental operation of low-pass filters in image processing.

Low-pass filters are designed to remove high-frequency components from an image while preserving low-frequency components. High-frequency components typically correspond to fine details, edges, and noise in the image, while low-frequency components represent the overall structure and general features. When a low-pass filter is applied to an image, it attenuates or blurs high-frequency components while allowing low-frequency components to pass through relatively unchanged. As a result, fine details and edges are smoothed out or blurred, leading to a reduction in image sharpness and clarity.

Example: Consider an image containing sharp edges and fine details, such as text on a signboard. When a low-pass filter, such as a Gaussian filter or a mean filter, is applied to this image, it suppresses the high-frequency components corresponding to the edges and fine details. As a result, the text on the signboard becomes less defined, and the edges appear smoother or blurred in the filtered image compared to the original image.

OR, Justify the statement, "Blurring of Images can significantly reduce noise". Explain with proper example.

The statement "Blurring of images can significantly reduce noise" can be justified based on the principle of noise reduction through spatial averaging.

When an image contains noise, such as random variations in pixel values, blurring the image effectively averages out these variations over a larger area, leading to a reduction in the overall noise level. This is because blurring involves smoothing neighboring pixels, which tends to diminish the impact of individual noisy pixels.

Example: Consider an image captured in low-light conditions, resulting in noticeable grainy noise. When a Gaussian blur filter is applied to this image, the filter averages the pixel values within a specified neighborhood for each pixel. As a result, the random noise in the image is smoothed out, leading to a reduction in the grainy appearance.

2(b):

A 5x5 bits/pixel original image is given by (4 bits/pixel)

15	12	8	9	14
12	12	12	14	11
13	13	10	9	10
15	12	10	12	11
13	14	13	13	14

i. Apply histogram equalization to the image by rounding the resulting image pixels to integer.
ii. Sketch the histograms of the original image and the histogram-equalized image.
iii. Why histogram equalization not produce a perfectly flat histogram?

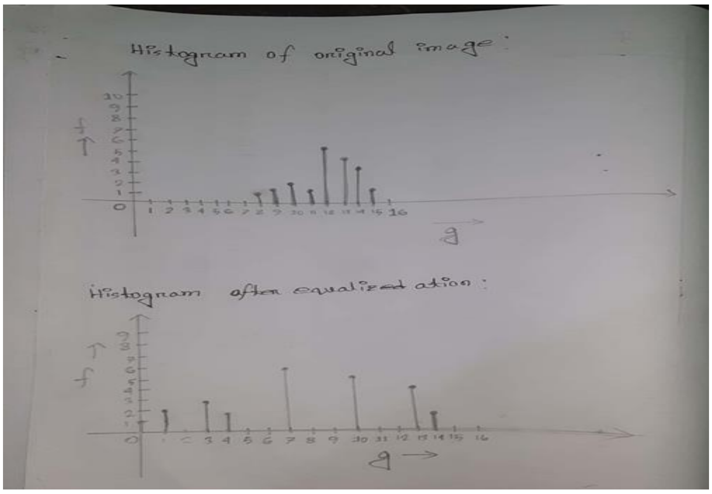
max - value = 15
∴ $2^4 = 16 > 15$
∴ $L = 16$; interval = [0, 15]

histogram equalization:
$N(g) = \max\{0, \text{round}(\frac{2^l \cdot c(g)}{n}) - 1\}$
$n = \Sigma f = 25$
$2^l = 16$

g	f	c	N	New-frequency
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	1	1	0	1
9	2	3	1	2
10	3	6	3	3
11	2	8	4	2
12	6	14	7	6
13	5	19	10	5
14	4	23	13	4
15	2	25	14	2

$\Sigma f = 25$

2(c)(ii): Answer



2(c)(iii): why histogram equalization not produce a perfectly flat histogram?

Answer: Histogram equalization redistributes pixel intensities in an image to achieve a more uniform histogram, enhancing the overall contrast and dynamic range. However, it does not always produce a perfectly flat histogram due to the non-uniform distribution of pixel intensities in the original image. In cases where the pixel intensities are already evenly distributed, histogram equalization may result in a flatter histogram, but in many real-world scenarios, there are variations in pixel intensities that prevent the histogram from becoming perfectly flat after equalization.

.

2(c):

A 4 x4 original image is given with 3 bits/pixel. 3

2	3	3	1
0	7	1	2
2	6	6	3
0	2	4	1

i. Perform Median filtering of the above image. (Use padding if necessary)

ii. "Performance of Median filtering is better than Averaging filtering" - Explain

https://www.youtube.com/watch?v=eJx3g-ZEfm4

The performance of median filtering is often considered better than average filtering due to its ability to effectively preserve edges and fine details while reducing noise. Median filtering replaces the central pixel value with the median value of neighboring pixels, making it robust against outliers and impulsive noise, such as salt-and-pepper noise. In contrast, average filtering computes the average intensity of neighboring pixels, which can blur edges and details, particularly in images with high-frequency content. Therefore, median filtering is preferred in scenarios where preserving image details is important, such as in medical imaging or surveillance applications.

3(a): What is the difference between spatial domain and frequency domain enhancement?

Spatial Domain Enhancement:

1. Operates directly on the pixel values of the image.
2. Techniques include spatial filtering (e.g., smoothing, sharpening) and point operations (e.g., histogram equalization).
3. Image enhancement is performed in the spatial (pixel) domain without transforming the image into a different domain.

Frequency Domain Enhancement:

1. Involves transforming the image from the spatial domain to the frequency domain using techniques like Fourier Transform.
2. Enhancements are performed in the frequency domain by manipulating frequency components.
3. Techniques include filtering in the frequency domain (e.g., high-pass, low-pass filtering) and frequency-based operations (e.g., notch filtering).

After enhancement, the image is transformed back to the spatial domain for visualization.

3(b): Write a short note with your own word to explain how do human beings perceive color? Given a color Image represented in terms of RGB components, how are the corresponding CMY and HIS coordinates derived?

Human beings perceive color through the eyes and the brain's interpretation of different wavelengths of light. The human eye contains specialized cells called cones that are sensitive to short (blue), medium (green), and long (red) wavelengths of light. The brain processes the signals from these cones to create the perception of a full spectrum of colors. When representing a color image in terms of RGB (Red, Green, Blue) components, the corresponding CMY (Cyan, Magenta, Yellow) and HIS (Hue, Intensity, Saturation) coordinates can be derived as follows:

CMY Coordinates:

Cyan (C) is the complement of Red (R), calculated as C = 1 - R.
Magenta (M) is the complement of Green (G), calculated as M = 1 - G.
Yellow (Y) is the complement of Blue (B), calculated as Y = 1 - B.

HIS Coordinates:

Hue (H): Represents the dominant color wavelength. It is calculated based on the RGB values, but the exact conversion can be complex. Various methods exist, such as using trigonometric functions to determine the angle in a color wheel.
Intensity (I): This represents the brightness of the color and is calculated as I = (R + G + B) / 3.
Saturation (S): Represents the purity of the color and is calculated as S = 1 - (3 / (R + G + B)) * min (R, G, B).

Converting RGB to CMY and HIS provides alternative color representations, each with its own interpretation and advantages for various applications, such as printing, image processing, and color analysis.

3(c):

3(c) A 4 x4 original image is given with 3 bits/pixel. 5

2	3	4	0
1	7	5	2
2	7	6	5
2	1	3	1

i) Perform Prewitt and Sobel operator on the image (Use padding)

ii) Analyze the differences of both images.

To perform Prewitt and Sobel operators on the given image, we first need to pad the image with zeros to avoid losing pixels at the edges. A common method is to add a border of zeros of size 1 pixel to each side of the image. After padding, the image becomes:

```
0 0 0 0 0 0 0
0 2 3 4 0 0 0
0 1 7 5 2 0 0
0 2 7 6 5 0 0
0 2 1 3 1 0 0
0 0 0 0 0 0 0
```

i. Prewitt and Sobel operators are used for edge detection in images. They are defined by two 3x3 masks, which are applied to each pixel in the image.

The Prewitt masks are:

```
      |-1 0 1|        |-1 -1 -1|
Px =  |-1 0 1|   Py = | 0  0  0|
      |-1 0 1|        | 1  1  1|
```

The Sobel masks are:

```
      |-1 0 1|        |-1 -2 -1|
Sx =  |-2 0 2|   Sy = | 0  0  0|
      |-1 0 1|        | 1  2  1|
```

To apply these masks, we convolve each mask with the image using a sliding window technique. The result is a new image where each pixel value represents the strength of the edge at that location.

Now, after applying the Prewitt operator, we get the following image:

```
2  4  4  1  0 0 0
-1 2  4  2 -2 0 0
1  4  6  4  3 0 0
3  4  4 -1 -4 0 0
0  1 -1 -2  0 0 0
0  0  0  0  0 0 0
```

After applying the Sobel operator, we get the following image:

```
3  5  3 -1 -1 0 0
-1 2  5  3 -3 0 0
0  3  6  5  2 0 0
3  3  2 -2 -5 0 0
-1 0 -2 -3  0 0 0
0  0  0  0  0 0 0
```

ii. The main difference between the Prewitt and Sobel operators is the way they weight the neighboring pixels in the mask. The Sobel operator gives more weight to the pixels in the center of the mask, resulting in a more accurate edge detection. In contrast, the Prewitt operator gives equal weight to all pixels in the mask, resulting in a more robust detection of edges in noisy images. However, the Sobel operator is more computationally intensive than the Prewitt operator. In the given image, we can see that the Sobel operator produces stronger edges than the Prewitt operator, but also introduces more noise. The choice of which operator to use depends on the specific requirements of the application.

OR,

A 4 x4 original image is given with 3 bits/pixel.

1	3	4	0
0	7	4	2
2	6	7	4
1	2	3	1

i) Perform Lowpass and High-pass filter on the image separately (Use padding)

ii) Analyze the statement " Lowpass + Highpass = Original image".

Low-pass filtering: https://www.youtube.com/watch?v=cS5w-kD8dcE&t=0s || High-pass filtering:

i. Performing lowpass filter on the given image using a 3x3 averaging filter:
1/9 * (1+3+4+0+0+7+4+2+2) = 2.4
1/9 * (1+3+4+0+0+7+4+2+6) = 3.0
1/9 * (1+3+4+0+0+7+4+2+7) = 3.333
1/9 * (0+7+4+2+6+7+3+1+3) = 3.111
1/9 * (0+7+4+2+6+7+3+1+1) = 2.666
1/9 * (2+6+7+4+2+3+1+3+1) = 3.0
1/9 * (1+2+3+1+0+0+2+6+7) = 2.111
1/9 * (0+7+4+2+6+7+3+1+3) = 3.111
1/9 * (0+7+4+2+6+7+3+1+1) = 2.666

The result of the lowpass filter is:

2 3 3 2
2 4 4 3
3 4 4 3
2 3 3 2

Performing high pass filter on the given image using 3x3 filter:

−1 −1 −1
−1 8 −1
−1 −1 −1

Applying the filter to the image (with padding to preserve size):
2+1+1+3+7+0+0-4+0 = 10
1-3-4-7+8+7+2-4+0 = 0
1-3-4-7+6+7+2-4-3 = -5
0-5-7-6+8+4+3-3-3 = -9
0-5-7-6+8+4+3-3-2 = -8
2-1-6-7+4+3+1-1-1 = -6
2-1-3+4+0+0+1-1-1 = 1
0-5-4+2+6+7+0-1-3 = 2
0-5-4+2+6+7+0-1-2 = 3

]The result of the high pass filter is:

10 0 −5 −9
0 −8 −6 −6
1 2 3 2
2 3 3 1

ii. The statement "Lowpass + Highpass = Original image" is a property of linear systems. In image processing, filtering is a linear operation, which means that the output of the filter is a linear combination of the input image pixels. Therefore, this property holds true for linear filters. A lowpass filter removes high-frequency components from the image, resulting in a smoother version of the original image. On the other hand, a highpass filter preserves high-frequency components

Au'22

2(a): Write the mathematical model of analog and digital image. Au'22 & sp'23

Mathematical model of an analog image: see above!!

Mathematical model for digital image:
An image may be defined as a two-dimensional function, f(x, y), where x and y are spatial (plane) coordinates, and the amplitude off at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point. When x, y, and the amplitude values of f are all finite, discrete quantities, we call the image a digital image.

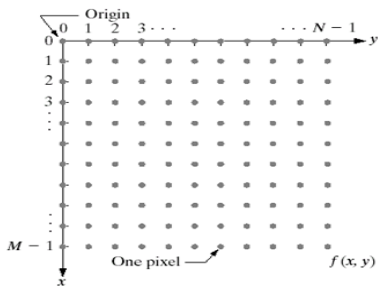


FIGURE 2.18 Coordinate convention used in this book to represent digital images.

2(b): Discuss the effects of reducing the spatial resolution of a digital image and effects of varying the number of intensity levels in a digital image. Give necessary example.

1. **Effects of reducing spatial resolution**:

Loss of Detail: Decreasing the spatial resolution reduces the number of pixels in the image, leading to a loss of fine details and sharpness.
Blurry Appearance: The image may appear blurry or pixelated, especially when viewing or printing at larger sizes.
Aliasing Artifacts: Reduced spatial resolution can introduce aliasing artifacts, such as jagged edges or moiré patterns, especially in images with high-frequency content.
Limited Enlargement: Images with lower spatial resolution may not scale well or may lose quality when enlarged.

Example: Consider an original image of a landscape with a spatial resolution of 3000 x 2000 pixels. After reducing the spatial resolution to 1500 x 1000 pixels, the image loses fine details such as individual leaves on trees or textures in the grass, resulting in a less detailed and less sharp appearance.

2. **Effects of varying the number of intensity levels**:

Loss of Color Gradient: Reducing the number of intensity levels decreases the color gradient or tonal range in the image, leading to banding or abrupt transitions between shades.
Posterization: Decreasing the number of intensity levels can cause posterization, where smooth gradients are replaced by abrupt changes in color or tone, resulting in a less realistic appearance.
Reduced Dynamic Range: Fewer intensity levels limit the dynamic range of the image, resulting in less subtle variations in brightness or color.
Loss of Contrast: The image may appear flat or lacking in contrast due to reduced intensity levels.

Example: Consider an original grayscale image with 256 intensity levels. After reducing the number of intensity levels to 64, the smooth transitions between shades of gray become more pronounced, leading to noticeable banding or posterization in areas with subtle gradients, such as skies or shadows.

Overall, both reducing spatial resolution and varying the number of intensity levels can significantly impact the visual quality and interpretability of digital images, leading to loss of detail, degradation in appearance, and potential artifacts.

2(c): How gray level slicing enhance the image? Why monotonically increasing function is used in special operation.

Gray level slicing is a technique used in image processing to enhance specific features or regions of interest within an image by manipulating the intensity levels of pixels. It involves selectively highlighting or isolating certain intensity ranges while suppressing others.

Enhancement with gray level slicing works as follows:

1. Selecting a range of intensity levels: A specific range of intensity levels is chosen based on the desired enhancement criteria. This range typically corresponds to the intensity levels associated with the features or regions to be emphasized.

2. Mapping intensity levels: Intensity levels within the selected range are mapped to higher values, while intensity levels outside the range are either left unchanged or mapped to lower values. This amplifies the contrast within the selected range and enhances the visibility of the targeted features.

3. Output image generation: The modified intensity values are used to generate the output image, where the emphasized features appear more pronounced compared to the surrounding areas.

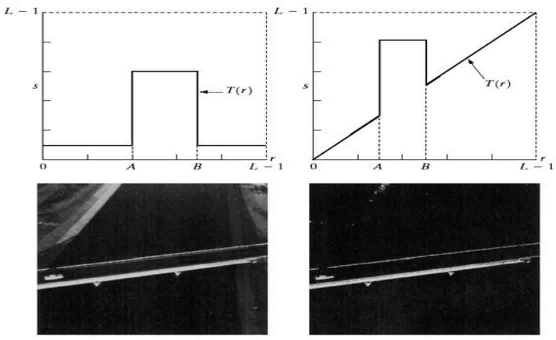
.

Highlighting a specific range of gray levels in an image. Two approaches:
t1, t2 = threshold value AB

g(x,y) = [if f(x,y) <A and if A<=f(x,y) <=B and if f(x,y) > B]
A, B = threshold value

g(x,y) = [if f(x,y) <A and if A<=f(x,y) <=B and if f(x,y) > B]

Gray level slicing can enhance specific features or regions in an image by selectively adjusting their contrast without affecting the rest of the image.



a b
c d

FIGURE 3.11 (a) This transformation highlights range [A, B] of gray levels and reduces all others to a constant level. (b) This transformation highlights range [A, B] but preserves all other levels. (c) An image. (d) Result of using the transformation in (a).

.

A monotonically increasing function is used in special operations like gray level slicing to ensure that the enhancement is consistent and preserves the relative ordering of intensity levels. By using a monotonically increasing function, the mapping of intensity levels maintains the original order, ensuring that the relationships between different intensity levels remain intact. This helps prevent artifacts or distortions that could arise from non-monotonic mappings, ensuring that the enhanced image accurately represents the desired features or regions of interest.

OR, is Monochrome image more suitable for image segmentation, then color image? Why?

Yes, monochrome (black and white or grayscale) images are often more suitable for image segmentation compared to color images. This is because monochrome images represent intensity variations in a single channel, simplifying the segmentation process. In contrast, color images introduce additional complexity due to the multiple channels (e.g., red, green, and blue), making it more challenging to define clear boundaries between objects. Monochrome images provide a simpler and more uniform representation, making it easier for segmentation algorithms to identify and separate objects based on intensity variations.

2(d): "In digital image processing high color image is presented by 12 bits where 24 bit is presented by true color"-justify the statement.

The statement highlights the difference between high color images, typically represented using 12 bits per pixel, and true color images, which are represented using 24 bits per pixel.

1. **High Color Images** (12-bit):
- High color images are represented using 12 bits per pixel.
- With 12 bits per pixel, there are 2^12 (or 4096) possible color combinations.
- In a high color image, each pixel is typically composed of 4 bits for the red channel, 4 bits for the green channel, and 4 bits for the blue channel.
- This allows for a wide range of colors to be represented, but not as wide as true color images.

2. **True Color Images (24-bit)**:
- True color images are represented using 24 bits per pixel.
- With 24 bits per pixel, there are 2^24 (or 16,777,216) possible color combinations.
- In a true color image, each pixel is typically composed of 8 bits for the red channel, 8 bits for the green channel, and 8 bits for the blue channel.
- This provides a much larger gamut of colors and allows for more accurate representation of real-world colors compared to high color images.

Justification:

- ***High color images,*** with 12 bits per pixel, offer a large number of color combinations but are limited compared to true color images in terms of the range of colors that can be represented.
- ***True color images,*** with 24 bits per pixel, provide a much wider range of color possibilities and are capable of accurately representing a vast array of colors found in real-world scenes.
- **Therefore,** while high color images offer a good compromise between color representation and file size, true color images provide the highest level of color accuracy and fidelity.

OR, In industrial applications for detecting missing components in product assembly which image processing can be used?

In industrial applications for detecting missing components in product assembly, image processing techniques such as:

Pattern Matching: Comparing captured images of assembled products against reference images to identify missing components based on predefined patterns or templates.

Edge Detection: Analyzing edges and contours within images to detect deviations or irregularities that may indicate missing components.

Thresholding: Setting intensity thresholds to distinguish between background and foreground components, allowing for the identification of missing parts based on intensity differences.

.

3(a): OR, suppose that a 3-bit image (L=8) of size 64\times64 pixels (MN=4096) has the intensity distribution in the figure 3, where the intensity levels are integers in the range [0.1,-1]=[0,7] Now sketch the original histogram, transformation function and equalized histogram.

Histogram $h(r_k) = n_k$

r_k is the k^{th} intensity value

n_k is the number of pixels in the image with intensity r_k

Normalized histogram $p(r_k) = \dfrac{n_k}{MN}$

n_k: the number of pixels in the image of size $M \times N$ with intensity r_k

Get the histogram equalization transformation function and give $p_s(s_k)$ for each s_k.

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

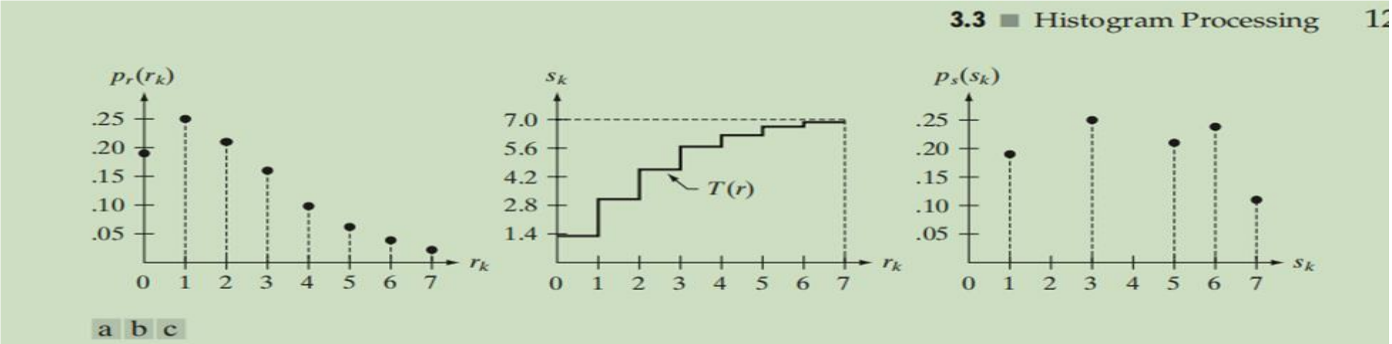
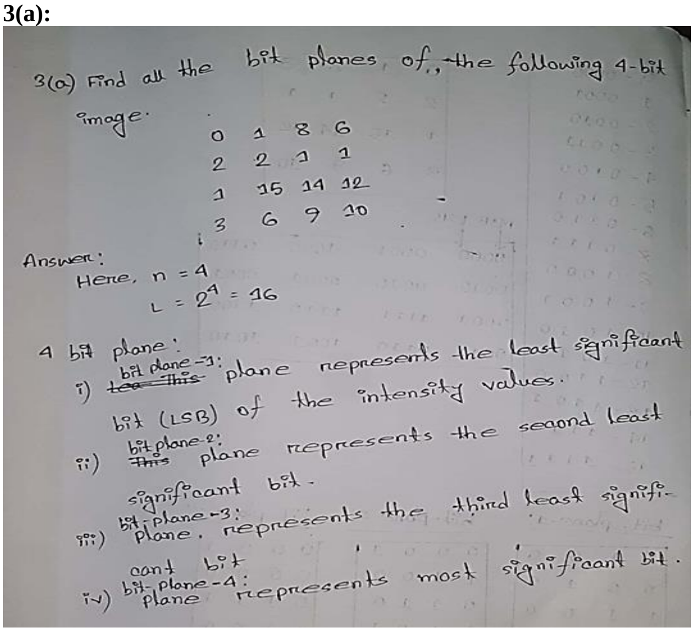


FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

3(a):



3(a) Find all the bit planes of the following 4-bit image.

0	1	8	6
2	2	1	1
1	15	14	12
3	6	9	10

Answer:
Here, n = 4
$L = 2^4 = 16$

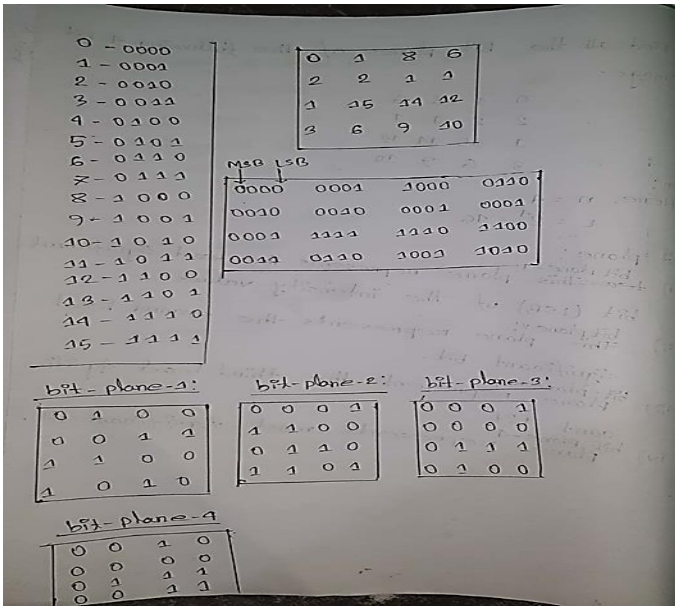
4 bit plane:

i) bit plane-1: ~~This~~ plane represents the least significant bit (LSB) of the intensity values.

ii) bit plane-2: ~~This~~ plane represents the second least significant bit.

iii) bit plane-3: plane represents the third least significant bit

iv) bit plane-4: plane represents most significant bit.



0 - 0000
1 - 0001
2 - 0010
3 - 0011
4 - 0100
5 - 0101
6 - 0110
7 - 0111
8 - 1000
9 - 1001
10 - 1010
11 - 1011
12 - 1100
13 - 1101
14 - 1110
15 - 1111

0	1	8	6
2	2	1	1
1	15	14	12
3	6	9	10

MSB LSB

0000	0001	1000	0110
0010	0010	0001	0001
0001	1111	1110	1100
0011	0110	1001	1010

bit-plane-1:

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

bit-plane-2:

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

bit-plane-3:

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

bit-plane-4

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

3(b): "The performance of Median filtering is better than low pass filtering for removing noise" - Why?

The performance of median filtering is often considered better than low-pass filtering for removing noise because median filtering effectively preserves edges and details in an image while effectively reducing impulsive noise, such as salt-and-pepper noise. Low-pass filtering, on the other hand, can blur the image and may not handle extreme noise values as well as median filtering does. Median filtering replaces the pixel value with the median of neighboring pixels, making it robust against outliers and more suitable for noise reduction without significantly compromising image details.

3(c): Explain spatial filtering in image enhancement. Explain different types of thresholding operations in short.

Spatial filtering in image enhancement involves applying a filter or mask to the pixels of an image in order to modify their values based on the values of neighboring pixels. This technique is used to enhance specific features or properties of an image, such as sharpening edges, reducing noise, or smoothing the image. Spatial filters typically involve moving a window or kernel over each pixel in the image and computing a new pixel value based on the values of the pixels within the window.

Types of spatial filtering operations in image enhancement include:

Smoothing Filters: These filters are used to reduce noise and blur the image by averaging the pixel values within the filter window. Examples include the Gaussian filter and the mean filter.

Sharpening Filters: These filters enhance edges and fine details in the image by accentuating intensity variations. Examples include the Laplacian filter and the Sobel filter.

Edge Detection Filters: These filters highlight edges and boundaries in the image by detecting rapid intensity changes. Examples include the Prewitt filter and the Roberts filter.

Noise Reduction Filters: These filters are designed to remove specific types of noise from the image, such as salt-and-pepper noise. The median filter is commonly used for this purpose.

Thresholding operations in image processing involve dividing an image into regions based on pixel intensity values and assigning them to different classes or categories. Different types of thresholding operations include: