

Prev Yr answer Image processing

Q. What is meant by image enhancement? What is contrast stretching? write an algorithm to explain how this operation is implement on a digital image?

Ans: Image enhancement is the process of improving the quality of an image to make it more visually appealing or easier to interpret. This can be done for a variety of reasons, such as to improve the visibility of certain features in the image, to reduce noise, or to correct for lighting or contrast problems.

Contrast stretching is a technique used to improve the contrast of an image. This can be done by expanding the range of pixel values in the image, or by making the differences between pixel values more pronounced.

Here the simple algorithm to implement contrast stretching on a digital image:

1. Input: Input image, represented as a matrix of intensity values (grayscale).
2. Compute Minimum and Maximum Intensity Values.
3. Contrast Stretching.
4. Output: `stretched_image = contrast_stretching(original_image)`.

Q. How does gray level slicing differ from bit plane slicing?

Ans: Gray level slicing and bit plane slicing are both techniques used to simplify images by removing certain levels of detail. However, they work in different ways.

Gray level slicing is a process of selecting a range of gray levels and setting all other pixels to black or white. This can be used to isolate certain features in an image, such as edges or textures. For example, if you want to isolate the edges of an object in an image, you could select a range of gray levels that correspond to the edges of the object.

Bit plane slicing is a process of breaking down an image into its individual bit planes. Each bit plane represents a different level of detail in the image. For example, the most significant bit plane (MSB) represents the most important details in the image, while the least significant bit plane (LSB) represents the least important details. Bit plane slicing can be used to compress images or to remove noise from images.

In general, gray level slicing is a more selective process than bit plane slicing. This is because gray level slicing allows you to specify the exact range of gray levels that you want to keep, while bit plane slicing removes all of the bits in a particular bit plane, regardless of their value.

Q. Why is "gamma" correction important in displaying an image accurately on a computer screen?

Ans: Gamma correction is crucial for accurately displaying images on computer screens due to the inherent non-linear relationship between the input voltage sent to the monitor and the resultant light output. This nonlinearity, known as gamma, causes the monitor to perceive and display darker tones differently from brighter tones. Gamma correction compensates for this non-linearity by adjusting the input voltage to achieve a consistent and accurate representation of the image's brightness and contrast across the entire range of tones. Without gamma correction, images would appear washed out or overly dark, depending on the specific monitor's gamma value.

Q. Suppose that a digital image is subjected to histogram equalization, what will be the result of second pass of histogram equalization? Explain with proper justification?

Ans: If a digital image is subjected to histogram equalization, the result of a second pass of histogram equalization will be the same as the first pass. This is because histogram equalization maps the pixel values of an image to a new range of values in a way that redistributes the pixel values so that the histogram of the image is more uniform. If the image is already histogram-equalized, then the pixel values are already in the most uniform possible distribution, so there is no need to re-map them.

Justification:

Histogram equalization is a technique that transforms the intensity distribution of an image to achieve a uniform intensity distribution. It aims to improve the contrast of an image by redistributing the pixels based on their intensity values.

When histogram equalization is applied to an image, the image's histogram is first calculated. The histogram represents the frequency distribution of pixel intensities in the image. After calculating the histogram, a cumulative distribution function (CDF) is derived. The CDF maps each pixel intensity to a cumulative probability value, which represents the percentage of pixels in the image that have a lower or equal intensity than the given pixel intensity.

During histogram equalization, a mapping function is created based on the CDF. This mapping function maps each pixel intensity to a new intensity value based on the CDF. The new intensity values are chosen in such a way that the resulting histogram of the image is approximately uniform.

If a second pass of histogram equalization is applied to an image that has already been histogram-equalized, the mapping function will remain the same. This is because the CDF of the already histogram-equalized image is already uniform, so there is no need to re-map the pixel values. As a result, the second pass of histogram equalization will not change the image.

Therefore, the second pass of histogram equalization on a histogram-equalized image will produce the same result as the first pass.

Q. What is meant by histogram matching? write down the different steps to be performed for histogram matching operation?

Ans: Histogram matching is a technique in image processing that aims to match the histogram of one image (reference image) to the histogram of another image (source image). This process essentially transforms the intensity distribution of the source image to resemble that of the reference image. Histogram matching is often employed to enhance the visual appearance of an image, improve image contrast, and correct color variations.

Steps for Histogram Matching:

1. Calculate Histograms: Compute the histograms of both the reference image and the source image. A histogram represents the frequency distribution of pixel intensities in an image.

2. Determine Cumulative Distribution Functions (CDFs): Derive the CDFs for both the reference image and the source image. A CDF maps each pixel intensity to a cumulative probability value, indicating the proportion of pixels with an intensity less than or equal to the given intensity.

3. Create a Mapping Function: Construct a mapping function based on the CDFs of the reference image and the source image. This function maps each pixel intensity in the source image to a corresponding intensity value in the reference image, ensuring that the CDFs align.

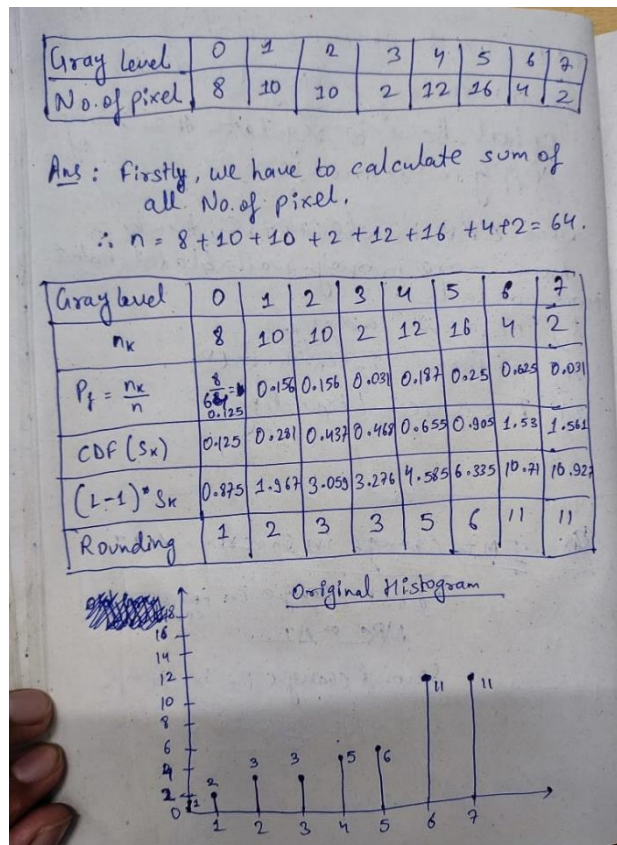
4. Apply the Mapping Function: Apply the mapping function to the source image to transform the pixel intensities. This transformation aligns the intensity distribution of the source image with that of the reference image.

5. Output the Transformed Image: The resulting image, with its transformed pixel intensities, is the histogram-matched image.

c) Perform histogram equalization on the following 8×8 image whose gray level distribution is given in the following table

Gray level	0	1	2	3	4	5	6	7
Number of pixels	8	10	10	2	12	16	4	2

Ans:



3. a) Compress ABRACADABRA using Huffman encoding. Using RLE, a string is compressed to 6A5C2B3D5E - what is the compression ratio?

Run length Encoding (RLE):

- very simple form of data compression.
- Applied in sequences in which the same data value occurs in many consecutive data elements.

let sequence of 6 characters 'A' \rightarrow 6A.

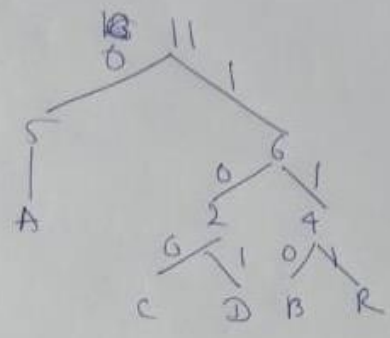
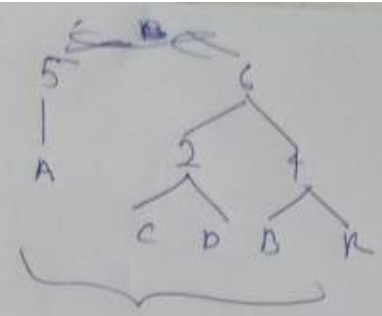
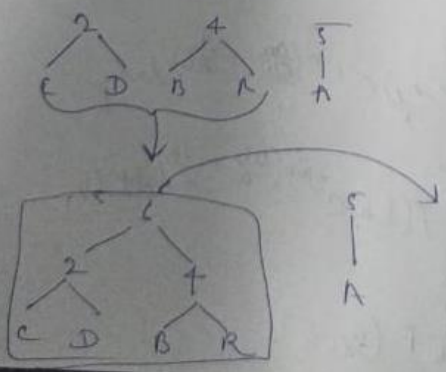
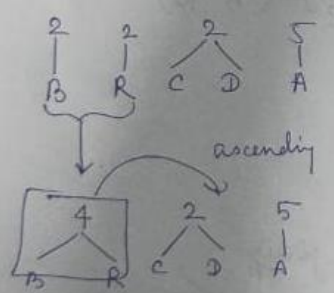
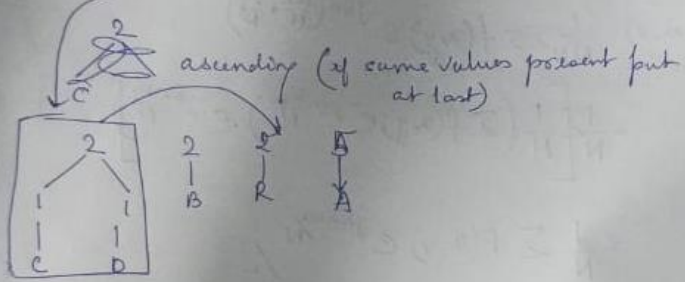
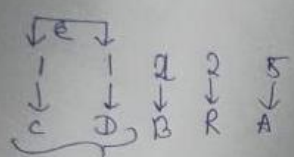
AAAAAACCCCCBBDDDEEEEEE can be represented as

6A5C2B3D5E.

Compression ratio: $21/10 = 2.1$.

$A \rightarrow 5$
 $B \rightarrow 2$
 $R \rightarrow 2$
 $C \rightarrow 1$
 $D \rightarrow 1$

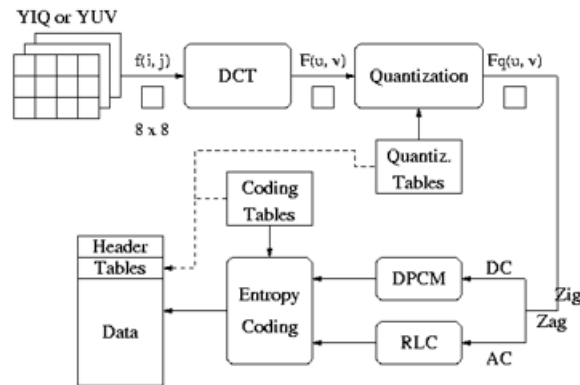
~~A~~~~B~~~~A~~~~C~~~~A~~~~D~~~~A~~~~R~~~~A~~



$A \rightarrow 0$
 $B \rightarrow 100$
 $D \rightarrow 101$
 $R \rightarrow 110$
 $R \rightarrow 111$

b) Draw the block diagram of JPEG image compression technique. Which step/s of JPEG image compression is/are responsible for lossy nature of JPEG? –explain with proper justification.

JPEG image compression-Block diagram



Step 1: Color Transformation (YCbCr)

This step converts the RGB color space to the YCbCr color space. YCbCr is a color space that separates the luminance (brightness) of an image from its chrominance (color). This separation allows for more efficient compression of the chrominance components, which can be discarded or approximated without significantly affecting the visual quality of the image.

Justification:

Discarding or approximating chrominance information results in a loss of color information, which is the main cause of lossiness in JPEG image compression.

Step 2: Discrete Cosine Transform (DCT)

This step applies the DCT to the YCbCr components of the image. The DCT is a mathematical operation that decomposes a signal into a series of orthogonal cosine functions. This decomposition allows for efficient compression of the image data by representing the image with a smaller number of coefficients.

Justification:

The DCT introduces quantization error, which is the difference between the original image values and the approximated values obtained from the DCT coefficients. This error is the main source of lossiness in JPEG image compression.

Step 3: Quantization

This step quantizes the DCT coefficients. Quantization involves dividing each DCT coefficient by a quantization step size and rounding the result to the nearest integer. This reduces the precision of the coefficients, which results in a loss of image quality.

Justification:

Quantization is the most lossy step in the JPEG compression process. The greater the quantization step size, the greater the loss of image quality.

Step 4: Entropy Coding

This step encodes the quantized DCT coefficients using entropy coding techniques such as run-length encoding and Huffman coding. Entropy coding is used to reduce the redundancy in the data, which further improves the compression ratio.

Justification:

Entropy coding does not introduce any loss of information, but it can make the compressed data more difficult to reconstruct.

4. a) Derive the discrete form of Laplacian operator (for image enhancement) from second order derivative.

Use of Second order derivatives for Enhancement - The Laplacian

→ Laplacian Filter → $\Delta^2 f$

Definition

$$\Delta^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

In x-direction

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$$

In y-direction

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

∴ $\Delta^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$

Filter:

$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$

$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$

0	1	0
1	-4	1
0	1	0

Different filters:

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

EC Academy

- b) Explain the output of using 3×3 average filter twice over an image.
 c) "Suppose an image contains salt and pepper noise" - which one of mean and median filtering, will produce better result? Give proper justification.

b). Applying a 3x3 average filter twice over an image in image processing results in a smoother image with reduced noise and sharper edges. The first pass of the filter removes high-frequency noise and blurs the image slightly. The second pass of the filter further smooths the image and enhances the edges.

Here's a more detailed explanation of the process:

First Pass:

When a 3x3 average filter is applied to an image, it replaces each pixel with the average of its 8 neighbouring pixels. This helps to reduce high-frequency noise, which is often caused by dust, sensor noise, or other imperfections in the image acquisition process. However, it also blurs the image slightly, as sharp edges are smoothed out by the averaging process.

Second Pass:

Applying the 3x3 average filter a second time further smooths the image and enhances the edges. The smoothing effect is stronger in areas with low contrast, while the edge enhancement effect is more pronounced in areas with high contrast. This is because the averaging process tends to reduce the contrast between neighbouring pixels, which makes edges stand out more prominently.

c). Median filtering is generally considered to be more effective than mean filtering in removing salt-and-pepper noise. This is because the median filter is less sensitive to outliers, which are the pixels that have values that are significantly different from the surrounding pixels. Salt-and-pepper noise is characterized by the presence of these outliers, and the median filter is better able to preserve the overall structure of the image while removing these outliers.

Here's a more detailed explanation of why median filtering is better for removing salt-and-pepper noise:

Mean filtering replaces each pixel with the average of its neighbouring pixels. This can be effective for removing low-frequency noise, but it can also blur the image and make edges less visible. Salt-and-pepper noise, on the other hand, is high-frequency noise, and it is not well-suited to mean filtering.

Median filtering, on the other hand, replaces each pixel with the median of its neighbouring pixels. This is more robust to outliers than mean filtering, so it is better able to remove salt-and-pepper noise without blurring the image. Additionally, median filtering preserves the order of the intensities in the image, which can help to retain important details.

a). write an algorithm for iterative global threshold operation in image segmentation. How adaptive thresholding can minimize the drawbacks of global thresholding operating? b). Mention the properties of region-based segmentation. write an algorithm for region growing technique using splitting and merging operation.

Ans: **A. Iterative Global Thresholding Operation**

The iterative global thresholding operation is a method for image segmentation that involves repeatedly applying a global threshold to an image until the desired segmentation is achieved. The iterative process is necessary because the optimal global threshold may not be known in advance, and it may also vary depending on the specific image.

Here's an algorithm for iterative global thresholding operation:

1. Initialize the threshold: Set an initial threshold value, typically the average or median intensity of the image.
2. Segment the image: Apply the global threshold to the image, resulting in a binary image where pixels above the threshold are labeled as foreground and pixels below the threshold are labeled as background.
3. Calculate the mean intensities: Compute the average intensity of the foreground and background pixels in the segmented image.
4. Update the threshold: Set the new threshold value to the average of the mean intensities of the foreground and background pixels.
5. Repeat steps 2-4: Continue the process of segmenting the image, calculating mean intensities, and updating the threshold until the desired segmentation is achieved or the convergence criteria are met.

Here are some specific ways in which adaptive thresholding minimizes the drawbacks of global thresholding:

1. Adapts to local illumination: Adaptive thresholding can adjust the threshold value based on the local brightness of the image, ensuring that the threshold is appropriate for each region of the image.
2. Preserves details in low-contrast regions: Global thresholding may overlook important details in low-contrast regions due to the fixed threshold value. Adaptive thresholding, on the other hand, can lower the threshold in these regions to better capture the subtle intensity variations.
3. Handles objects with varying intensity: Global thresholding may struggle to segment objects that have varying intensity values across their extent.

Adaptive thresholding can adjust the threshold locally to better delineate such objects.

4. Reduces over-segmentation and under-segmentation: Global thresholding can lead to over-segmentation (too many regions) or under-segmentation (too few regions) due to the fixed threshold. Adaptive thresholding can help to prevent these issues by adjusting the threshold based on local image properties.

B. Adaptive Thresholding

Adaptive thresholding is a technique that aims to overcome the limitations of global thresholding by dynamically adjusting the threshold based on local image characteristics. This allows for more accurate segmentation of images with non-uniform illumination or varying contrast.

Properties of Region-Based Segmentation

Region-based segmentation algorithms group pixels based on their shared properties, such as intensity, texture, or color. These methods aim to identify and group together pixels that belong to the same object or region in the image.

Region Growing Technique Using Splitting and Merging Operation

The region growing technique is a common approach for region-based segmentation. It starts with an initial seed point and iteratively grows the region by merging neighboring pixels that satisfy certain criteria.

Here's an algorithm for region growing technique using splitting and merging operation:

1. Initialize the region: Select an initial seed point within the image and mark it as the current region.

2. Grow the region: Consider the neighboring pixels of the current region. Merge neighboring pixels into the region if they satisfy the criteria, such as similar intensity or texture.

3. Split the region: If the current region becomes too large or exhibits significant heterogeneity, split it into smaller sub-regions and repeat the growing process for each sub-region.

4. Continue the process: Repeat steps 2 and 3 until no more pixels can be merged or split, and the desired segmentation is achieved.

6. Write short notes on
 - a. Histogram stretching
 - b. Sampling and quantization
 - c. Weighted average filter
 - d. Spatial resolution enhancement by interpolation

a. Histogram stretching

Histogram stretching is a technique used to improve the contrast of an image by redistributing the pixel values. This is done by mapping the original pixel values to new pixel values in such a way that the cumulative distribution function (CDF) of the new pixel values is approximately uniform. This results in a more uniform distribution of intensities in the image, which can make the image appear more visually appealing.

b. Sampling and quantization

Sampling and quantization are two fundamental processes in digital image processing. Sampling is the process of converting a continuous image signal into a discrete representation. This is done by taking samples of the image at regular intervals. Quantization is the process of reducing the number of bits used to represent each sample. This is done by rounding the sample values to the nearest integer.

c. Weighted average filter

A weighted average filter is a type of spatial filter that replaces each pixel in an image with the weighted average of its neighbouring pixels. The weights determine the relative contribution of each neighbouring pixel to the new value. Weighted average filters are often used to smooth images and remove noise.

d. Spatial resolution enhancement by interpolation

Spatial resolution enhancement is the process of increasing the number of pixels in an image. This can be done using a variety of interpolation techniques, such as nearest neighbour interpolation, bilinear interpolation, and bicubic interpolation. Interpolation works by estimating the values of missing pixels based on the values of neighbouring pixels.