Q1. Describe the two categories of Image segmentation.

In digital image processing and computer vision, **image segmentation** is the process of partitioning a digital image into multiple **image segments**, also known as **image regions** or **image objects** (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze.[1][2] Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

Two types of image segmentation exist:

- **Semantic segmentation.** Objects shown in an image are grouped based on defined categories. For instance, a street scene would be segmented by “pedestrians,” “bikes,” “vehicles,” “sidewalks,” and so on.

Semantic segmentation | Keymakr

- **Instance segmentation.** Consider instance segmentation a refined version of semantic segmentation. Categories like “vehicles” are split into “cars,” “motorcycles,” “buses,” and so on — instance segmentation detects the instances of each category.

Instance segmentation | Keymakr

The difference between semantic vs. instance vs. panoptic segmentation lies in how they process the `things` and `stuff` in the image.

Semantic segmentation studies the uncountable `stuff` in an image. It analyzes each image pixel and assigns a unique class label based on the texture it represents. For example, in **Figure 1**, an image contains two cars, three pedestrians, a road, and the sky. The two cars represent the same texture as do the three pedestrians.

Figure 1: Semantic segmentation examples (source: **SegNet**).

Semantic segmentation would assign unique class labels to each of these textures or categories. However, semantic segmentation's output cannot differentiate or count the two cars or three pedestrians separately. Commonly used semantic segmentation techniques include SegNet, U-Net, DeconvNet, and FCNs.

Instance segmentation typically deals with tasks related to countable `things`. It can detect each object or instance of a class present in an image and assigns it a different mask or bounding box with a unique identifier.

For example, instance segmentation would identify the two cars in the previous example separately as, let's say, `car_1` and `car_2`. Commonly used instance segmentation techniques are Mask R-CNN, Faster R-CNN, PANet, and YOLACT. **Figure 2** demonstrates different instance segmentation detections.

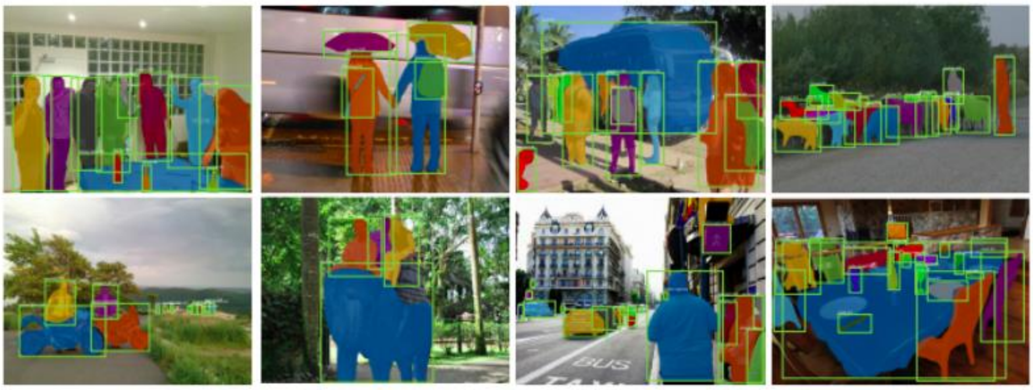


Figure 2: Instance segmentation examples (source: Mask R-CNN).

The goal of both semantic and instance segmentation techniques is to process a scene coherently. Naturally, we want to identify both `stuff` and `things` in a scene to build more practical real-world applications. Researchers devised a solution to reconcile both `stuff` and `things` within a scene (i.e., panoptic segmentation).

Q2.

Briefly explain the different approaches for image segmentation.

Types of Image Segmentation

1. The Approach

- Whenever one tries to take a bird's eye view of the Image Segmentation tasks, one gets to observe a crucial process that happens here – object identification.

1.1. Similarity Detection (Region Approach)

- This fundamental approach relies on detecting similar pixels in an image – based on a threshold, region growing, region spreading, and region merging.

1.2. Discontinuity Detection (Boundary Approach)

- This is a stark opposite of similarity detection approach where the algorithm rather searches for discontinuity. Image Segmentation Algorithms like Edge Detection, Point Detection, Line Detection follows this approach – where edges get detected based on various metrics of discontinuity like intensity etc.

Edge Vs Detection

Q3. Illustrate the segmentation methods based on points, lines and edges

3.1. Point Detection [8] [16]

A point is the most basic type of discontinuity in a digital image. The most common approach to finding discontinuities is to run an $(n \times n)$ mask over each point in the image. The mask is as shown in figure 2.

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

The point is detected at a location (x, y) in an image where the mask is centered. If the corresponding value of R such that

$$\| R \| > T \qquad (2)$$

Where R is the response of the mask at any point in the image and T is non-negative threshold value. It means that isola ted point is detected at the corresponding value (x, y). This formulation serves to measures the weighted differences between the center point and its neighbors since the gray level of an isolated point will be very different from that of its neighbors [ ]. The result of point detection mask is as shown in figure 3.

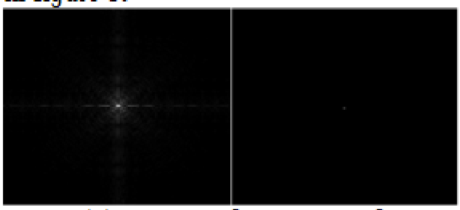


Figure 3. (a) Gray-scale image with a nearly invisible isolated black point (b) Image showing the detected point

3.2. Line Detection [8] [16]

Line detection is the next level of complexity in the direction of image discontinuity. For any point in the image, a response can be calculated that will show which direction the point of a line is most associated with. For line detection, we use two masks, i^{th} and, j^{th} mask. Then, we have

$$\| R_i \| > \| R_j \|, \ \forall j \neq i \qquad (3)$$

It means that the corresponding points is more likely to be associated with a line in the direction of the mask i.

Figure 4. Line Detector masks in (a) Horizontal direction (b) 45° direction (c) Vertical direction (d) - 45° direction

The greatest response calculation from these matrices will yield the direction of the given pixel []. The result of line detection mask is as shown in figure 5.

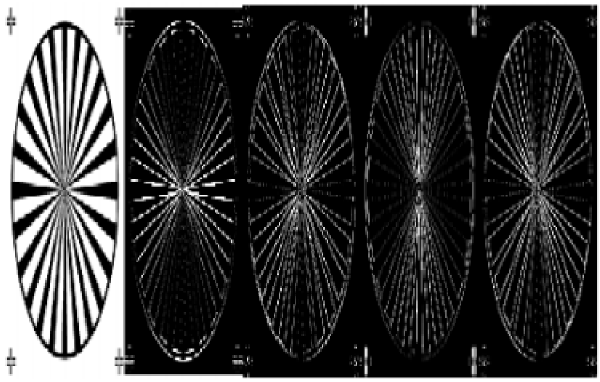


Figure 5. (a) Original Image (b) result showing with horizontal detector (c) with 45° detector (d) with vertical detector (e) with -45° detector

With the help of lines detector masks, we can detect the lines in a specified direction. For example, we are interesting in finding all the lines that are one pixel thick, oriented at -45°. For that, we take a digitized (binary portion of a wire-bond mask for an electronics circuit. The results are shown as in figure 6.

3.3. Edge detection [8] [16]

Since isolated points and lines of unitary pixel thickness are infrequent in most practical application, edge detection is the most common approach in gray level discontinuity segmentation. An edge is a boundary between two regions having distinct intensity level. It is very useful in detecting of discontinuity in an image. When the image changes from dark to white or vice-versa. The changes of intensity, first-order derivative and second-order derivative are shown in figure 7.

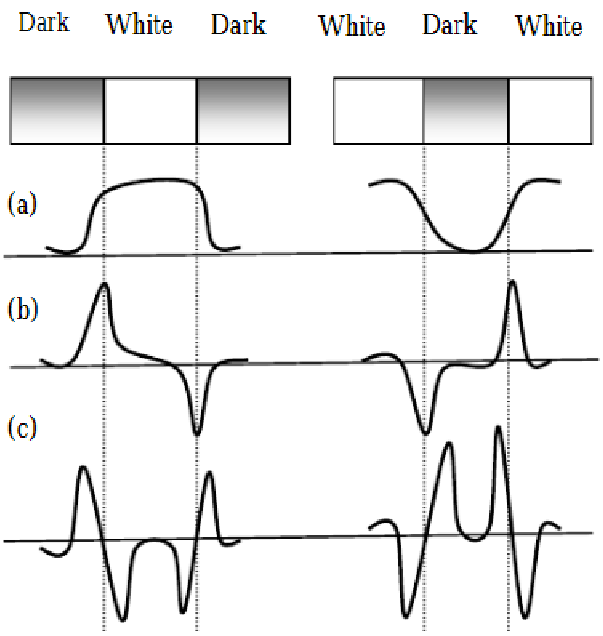


Figure 7. (a) Intensity profile (b) First-order derivatives (c) Second-order derivatives

3.3.1. First-order derivatives. First-order derivatives responds whenever there is discontinuity in intensity level. It is positive at the leading edge and negative at the trailing edge. Suppose we have an image f(x, y) and gradient operator $\vec{\nabla}$f.

$$f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \qquad (4)$$

Strength of $\vec{\nabla} f$ is given by

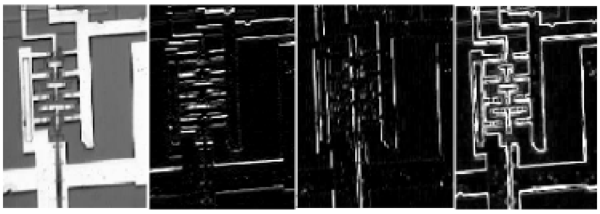
$$\begin{aligned} \nabla f &= \text{magnitude of } (\vec{\nabla} f) \\ &= \sqrt{G_x^2 + G_y^2} \\ &\propto \| G_x \| + \| G_y \| \end{aligned} \qquad (5)$$

It gives the strength of edge at location (x, y)

Direction of $\vec{\nabla} f$ is given by

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_x}{G_y}\right) \qquad (6)$$

Where $\alpha(x, y)$ gives the direction of $\vec{\nabla} f$.



(a) (b) (c) (d)

Figure 8. (a) Original Image (b) $\| G_x \|$ component of the gradient along x-direction (c) $\| G_y \|$ component of the gradient along y-direction (d) Gradient Image $\| G_x \| + \| G_y \|$

There is several ways to calculate the image gradient:

3.3.1.1. Prewitt Edge operator

-1	-1	-1
0	0	0
-1	-1	-1

-1	0	-1
-1	0	-1
-1	0	-1

Figure 9. Masks used for Prewitt Edge operator

The mask finds the horizontal edges is equivalent to gradient in the vertical direction and the mask compute the vertical edges is equivalent to gradient in the horizontal direction. Using these two masks passing to the intensity image, we can find out G_x and G_y component at different location in an image. So, we can find out the strength and direction of edge at that particular location (x, y).

3.3.1.2. Sobel Edge operator

-1	-2	-1	-1	0	1
0	0	0	-2	0	2
1	2	1	-1	0	1

Figure 10. Masks used for Sobel Edge operator

It gives the averaging effect over an image. It considers the effect due to the spurious noise in the image. It is preferable over prewitt edge operator because it gives the smoothing effect and by which we can reduce spurious edge which are generated because of noise present in the image.

3.3.2. Second-order derivatives

It is positive at the darker side and negative at the white side. It is very sensitive to noise present in an image. That's why it is not used for edge detection. But, it is very useful for extracting some secondary information i.e. we can find out whether the point lies on the darker side or the white side.
Zero-crossing: It is useful to identify the exact location of the edge where there is gradual transition of intensity from dark to bright region and vice-versa. There are several second-order derivative operators:

3.3.2.1. **Laplacian operator.** The Laplacian mask is given by:

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

Figure 11. Masks used for Laplacian operator

$$\nabla^2(f) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \qquad (7)$$

If we consider the diagonal elements:

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

Figure 12. Masks used for Laplacian operator using 8-connectivity

It is not used for edge detection because it is very sensitive to noise and also leads to double edge. But, it is very useful for extracting secondary information. To reduce the effect of noise, first image will be smooth using the Gaussian operator and then it is operated by Laplacian operator. These two operations together is called LoG (Laplacian of Gaussian) operator.

3.3.2.2. LoG operator

The LoG mask is given by

0	0	-1	0	0
0	-1	2	-1	0
-1	-2	1	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

Figure 13. Masks used for LoG operator

The Gaussian operator is given by:

$$h(x,y) = \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \qquad (8)$$

$$where\ x^2 + y^2 = r^2$$

$$\nabla^2 h = \left(\frac{r^2-\sigma^2}{\sigma^4}\right) exp\left(-\frac{r^2}{2\sigma^2}\right) \qquad (9)$$

3.3.2.3. Canny operator [4]

It is very important method to find edges by isolating noise from the image before find edges of images, without affecting the features of the edges in the image and then applying the tendency to find the q5. Describe any one algorithm used for edge detection.

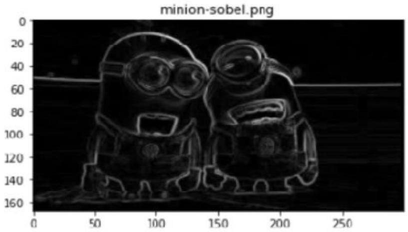
Sobel Edge Detection: This uses a filter that gives more emphasis to the centre of the filter. It is one of the most commonly used edge detectors and helps reduce noise and provides differentiating, giving edge response simultaneously. The following are the filters used in this method-

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Sobel filter for vertical edge detection **Sobel filter for horizontal edge detection**

The following shows the before and after images of applying Sobel edge detection-

original minion image



minion image after applying sobel edge detector

6. Define thresholding and explain the various techniques of thresholding in detail.

Image thresholding is a simple form of image segmentation. It is a way to create a binary image from a grayscale or full-color image. This is typically done in order to separate "object" or foreground pixels from background pixels to aid in image processing.

Image thresholding is the easiest way to separate image background and foreground. Also, this image thresholding can be identify as image segmentation. To apply thresholding techniques, we should use a gray scale image. When thresholding, that grayscale image will be converted to a binary image. In this article, we will discuss about different types of digital image thresholding techniques.

Simple Thresholding…

> *In simple thresholding, all pixel values those are greater than the specific threshold value, assign to standard value. It compares pixel values with special threshold value. After separating the pixel, we can see the segmented images according to threshold values. Threshold techniques are mainly divided in to 3 categories based on the threshold operator.*

- Global → threshold operator depends on the gray values of the pixel
- Local → threshold operator depends on the gray values of the pixel and local properties
- Dynamic → threshold operator depends on the gray values of the pixel and local properties and it's position

Basic Global Thresholding …

Global thresholding depends on the histogram of the image. This section again divides in to 2 categories : ***histogram improvement*** method and ***threshold computing*** method.

Histogram Improvement ...

Using this method, we can easily separate the image segments. But this method will fail when using images with complex backgrounds. Also, this method sensitive to noises.

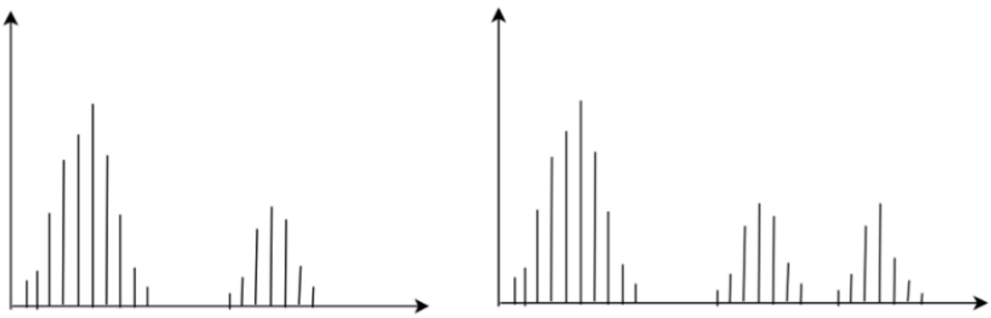


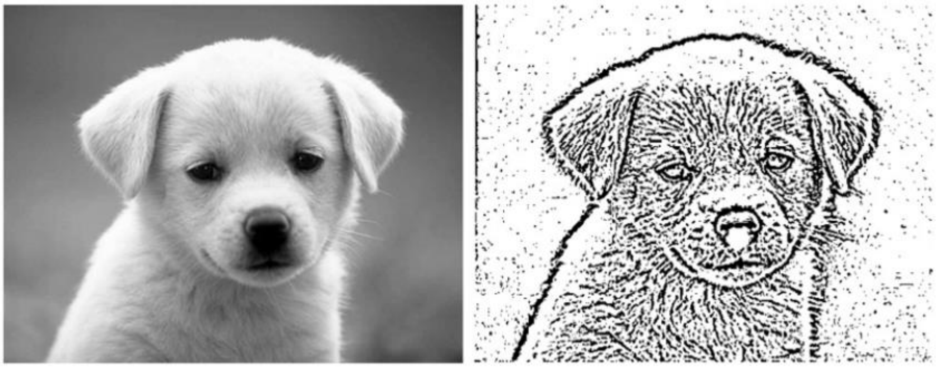
Image with dark background and 1 light object   Image with dark background and 2 light object

Threshold Computing...

Threshold computing methods can be use for complex backgrounds and multiple objects. It uses local both properties and gray level information for the thresholding process. Therefore , threshold computing methods is most effective than histogram improvement method. There are several threshold computing methods such as Ada 'ding, Otsu thresholding and Binary thresholding.

Adaptive Thresholding ...

> *In this technique, input image will be segmented in to small areas. These areas are non overlapping areas. Different threshold values are use to each image segment. Threshold values are depends on the locations of pixel values. Because, each image segment has different light areas.*



Truncate Thresholding …

In this method, the destination pixel will be set to the threshold value. Pixel values which are greater than the threshold value will be set to threshold value. Other values are remaining same.

Otsu Thresholding …

> *This "Automatic Optimal Threshold Detection" technique finds optimum global threshold value from histogram of the image. When image histogram has 2 peaks, optimal threshold value should be in the middle of those values. Image will be separated in to foreground and background according to threshold value. There are 2 categories in Otsu method called 1-D and 2-D. 2-D method is more effective than 1-D technique.*

Threshold to Zero …

> *In this technique, pixel value compares with the threshold value. If the value is grater than threshold value, then corresponding value will be assign. Otherwise, pixel value will be set to Zero.*

Threshold to zero

Threshold to zero inverted

q7. Discuss the different region based image segmentation techniques.

Region-Based Segmentation

In this segmentation, we grow regions by recursively including the neighboring pixels that are similar and connected to the seed pixel. We use similarity measures such as differences in gray levels for regions with homogeneous gray levels. We use connectivity to prevent connecting different parts of the image.

There are two variants of region-based segmentation:

- **Top-down approach**
  - First, we need to define the predefined seed pixel. Either we can define all pixels as seed pixels or randomly chosen pixels. Grow regions until all pixels in the image belongs to the region.
- **Bottom-Up approach**
  - Select seed only from objects of interest. Grow regions only if the similarity criterion is fulfilled.
- **Similarity Measures:**
  - Similarity measures can be of different types: For the grayscale image the similarity measure can be the different textures and other spatial properties, intensity difference within a region or the distance b/w mean value of the region.
- **Region merging techniques:**
  - In the region merging technique, we try to combine the regions that contain the single object and separate it from the background.. There are many regions merging techniques such as Watershed algorithm, Split and merge algorithm, etc**.**
- **Pros:**
  - Since it performs simple threshold calculation, it is faster to perform.
  - Region-based segmentation works better when the object and background have high contrast.
- **Limitations:**
  - It did not produce many accurate segmentation results when there are no significant differences b/w pixel values of the object and the background.

9. Define image feature? How are the features classified?

Features are parts or patterns of an object in an image that help to identify it.

Feature extraction is a process of dimensionality reduction by which an initial set of raw data is reduced to more manageable groups for processing. A characteristic of these large data sets is a large number of variables that require a lot of computing resources to process.

Feature Extraction is an important technique in Computer Vision widely used for tasks like:

- Object recognition
- Image alignment and stitching (to create a panorama)
- 3D stereo reconstruction
- Navigation for robots/self-driving cars

Classification

There are several types of classification:

Type of Classification	Description	Example
Categorical (Nominal)	Classification of entities into particular categories.	That thing is a dog. That thing is a car.
Ordinal	Classification of entities in some kind of ordered relationship.	You are stronger than him. It is hotter today than yesterday.
Adjectival or Predicative	Classification based on some quality of an entity.	That car is fast. She is smart.
Cardinal	Classification based on a numerical value.	He is six feet tall. It is 25.3 degrees today.

1)Character level features: Computerized handwriting identification consist of two type of handwriting features macro and micro. Macro features are gray scale based , contour based, slope stroke-width, slant and height.

2) The structural features representing the coarser shape of the character capture the presence of corners, diagonal lines, and vertical and horizontal lines in the gradient image.

3) The concavity features capture the major topological and geometrical features including direction of bays, presence of holes, and large vertical and horizontal strokes.

4) Micro features (GSC) are found to be discriminating for the writers. Furthermore, they found ’G’, ’b’, ’N’, ’I’, ’K’, ’J’,’W’, ’D’, ’h’, ’f’ are the 10 most discriminating characters.

10. Discuss the classification of feature extraction in Image processing

Image Feature Extraction class work

Part 2

11. Define Representation. Explain two types of Image representation with an example.

Abstract—Image Representation is the process of generating descriptions from the visual contents of an image. There is a wide range of image representation methods which have been proposed in the past few decades. Some of these methods are designed for specific application areas while others are more generalized and can be applied in various fields. Each of the image representation methods is unique in its own way and has its advantages and drawbacks. This paper does a brief review on different existing two dimensional image representation methods by focusing its approach, advantages, limitations and applications. The analysis shows that most of the representations is based on machine perception model which cannot be understandable by a human being just by looking through his eyes. The paper ends with the suggestion of introducing a representation which can understandable by both a human and machine.

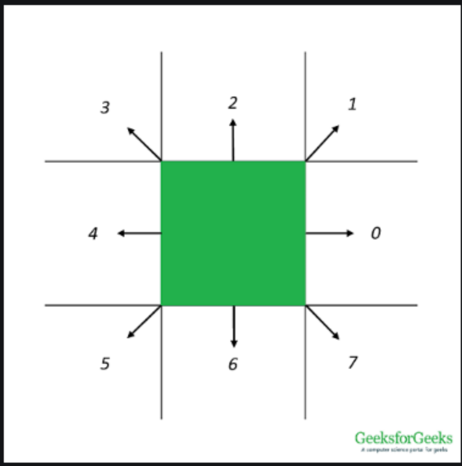
III. CLASSIFICATIONS ON IMAGE REPRESENTATIONS Image Representation is the foundation of good performance of various image processing tasks. To represent each image effectively a large number of image representation methods have been proposed over the time [8]. And these image representation methods can be classified on the basis of three parameters: (1) Based on level of processing (2) Based on level of abstraction (3) Based on image features.

3. Region based representations: Also known as superpixel representation. Here the regions are not rectangular and it is formed by grouping similar and connected pixels. The adjacency information between regions is represented usually as RAG(region-adjacency graph) or combinatorial map. The representation is used for object detection and segmentation, but different unions of multiple regions have to be considered [9]

12.Explain Chain Code with an example.

Chain code is a lossless compression technique used for representing an object in images. The co-ordinates of any continuous boundary of an object can be represented as a string of numbers where each number represents a particular direction in which the next point on the connected line is present. One point is taken as the reference/starting point and on plotting the points generated from the chain, the original figure can be re-drawn.

This article describes how to generate a 8-neighbourhood chain code of a 2-D straight line. In a rectangular grid, a point can have at most 8 surrounding points as shown below. The next point on the line has to be one of these 8 surrounding points. Each direction is assigned a code. Using this code we can find out which of the surrounding point should be plotted next.



The chain codes could be generated by using conditional statements for each direction but it becomes very tedious to describe for systems having large number of directions(3-D grids can have up to 26 directions). Instead we use a hash function. The difference in X(dx) and Y(dy) co-ordinates of two successive points are calculated and hashed to generate the key for the chain code between the two points.

Chain code list: $[5, 6, 7, 4, -1, 0, 3, 2, 1]$

Hash function: $C(dx, dy) = 3dy + dx + 4$

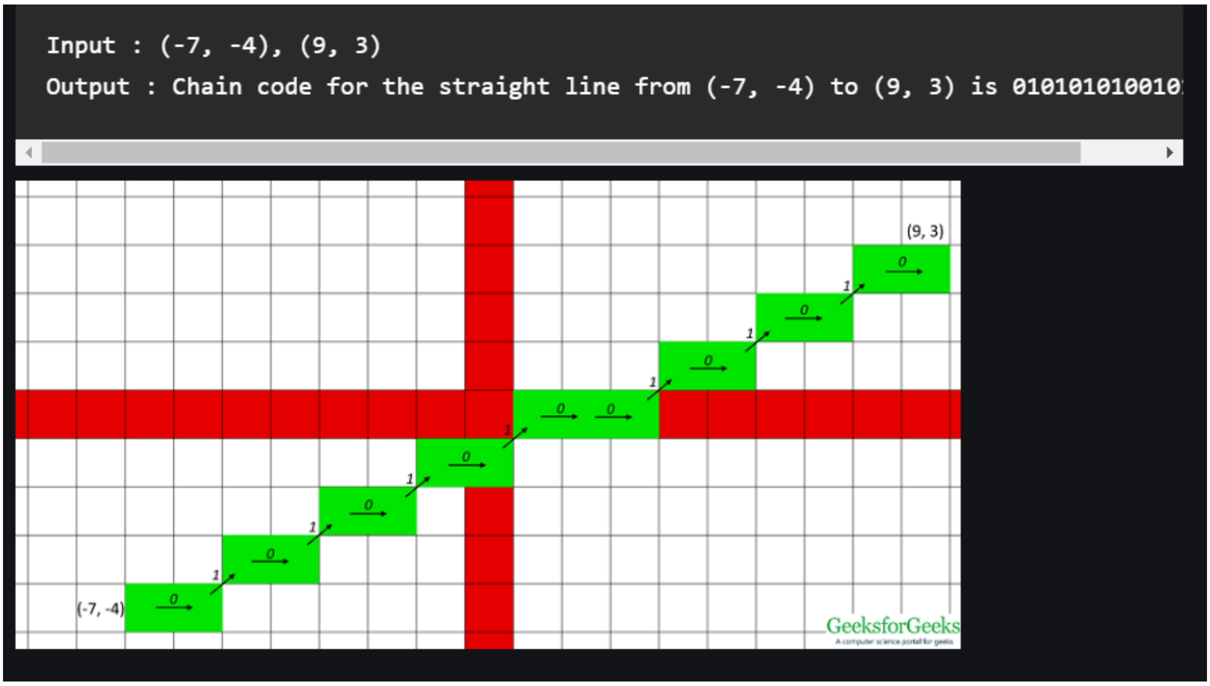
Hash table:-

dx	dy	$C(dx, dy)$	$chainCode[C]$
1	0	5	0
1	1	8	1
0	1	7	2
-1	1	6	3
-1	0	3	4
-1	-1	0	5
0	-1	1	6
1	-1	2	7

The function does not generate the value 4 so a dummy value is stored there.

Examples:

```
Input : (2, -3), (-4, 2)
Output : Chain code for the straight line from (2, -3) to (-4, 2) is 333433
```



13.Discuss Fourier Descriptors of the boundary. What are the properties of Fourier Descriptors?

Fourier descriptors A method used in object recognition and image processing to represent the boundary shape of a segment in an image. The first few terms in a Fourier series provide the basis of a descriptor. This type of object descriptor is useful for recognition tasks because it can be designed to be independent of scaling, translation, or rotation.

Fourier Transform: Fourier transform is the input tool that is used to decompose an image into its sine and cosine components.

Linearity:
Addition of two functions corresponding to the addition of the two frequency spectrum is called the linearity. If we multiply a function by a constant, the Fourier transform of the resultant function is multiplied by the same constant. The Fourier transform of sum of two or more functions is the sum of the Fourier transforms of the functions

Scaling:
Scaling is the method that is used to the change the range of the independent variables or features of data. If we stretch a function by the factor in the time domain then squeeze the Fourier transform by the same factor in the frequency domain.

Differentiation:
Differentiating function with respect to time yields to the constant multiple of the initial function.

Convolution:
It includes the multiplication of two functions. The Fourier transform of a convolution of two functions is the point-wise product of their respective Fourier transforms.

Frequency Shift:
Frequency is shifted according to the co-ordinates. There is a duality between the time and frequency domains and frequency shift affects the time shift.

Time Shift:
The time variable shift also effects the frequency function. The time shifting property concludes that a linear displacement in time corresponds to a linear phase factor in the frequency domain.

14.Write short notes on Image representation:

a. Using boundary descriptors b. Using regional descriptors

BOUNDARY DESCRIPTORS

Simple Descriptors

● Length of a Contour By counting the number of pixels along the contour. For a chain coded curve with unit spacing in both directions, the number of vertical and horizontal components plus 21/2 times the number of components give the exact length of curve

. ● Boundary Diameter It is defines as Diam (B) = max [D(pi , pj)] i, j where D is the distance measure which can be either Euclidean distance or D4 distance. The value of the diameter and the orientation of the major axis of the boundary are two useful Descriptors.

BOUNDARY DESCRIPTORS

● Curvature It is the rate of change of slope. Curvature can be determined by using the difference between the slopes of adjacent boundary segments at the point of intersection of the segments. Shape Numbers Shape number is the smallest magnitude of the first difference of a chain code representation. The order of a shape number is defined as the number of digits in its representation. Shape order is even for a closed boundary.

REGIONAL DESCRIPTORS Simple Descriptors Area, perimeter and compactness are the simple region Descriptors Compactness = (perimeter)2/area

Topological Descriptors ● Rubber-sheet Distortions Topology is the study of properties of a figure that are unaffected by any deformation, as long as there is no tearing or joining of the figure. ● Euler Number Euler number (E) of region depends on the number of connected components (C) and holes (H). $E = C - H$ A connected component of a set is a subset of maximal size such that any two of its points can be joined by a connected curve lying entirely within the subset.

https://www.ecb.torontomu.ca/~gnkhan/Computer-Vision/Lectures/Image-Rep.pdf link