INTRODUCTION:

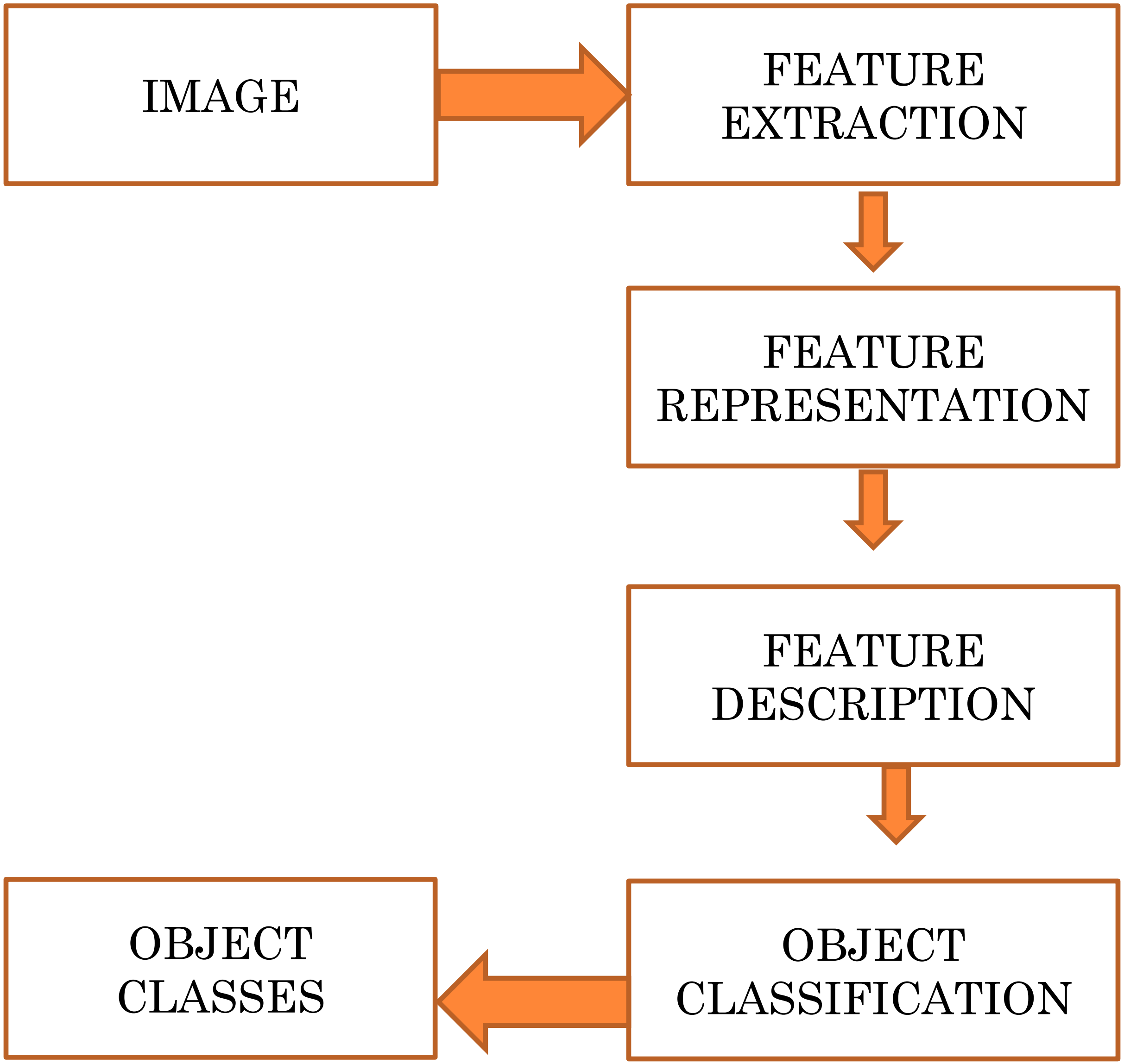
- Feature information of an image is extracted from its content.
- It may include format descriptors, intrinsic features, spatial features, relationships among image entities, etc.
- This information is further used for retrieval purpose of images.

- The requirement of feature information is mainly the function of its application domain.
- For example: the medical domain in particular, relative spatial (geometry) relationships among internal image entities contribute as a major factor for image similarity.

DEFINITION:

- Feature extraction is a process of transforming the input data (without redundancy or duplicate value) into set of features.
- Instead of full size input, the features will be extracted from the reduced data containing the relevant information.

- It involves using algorithms to detect and isolate various desired portions or shapes (features) of a digitized image or video stream.

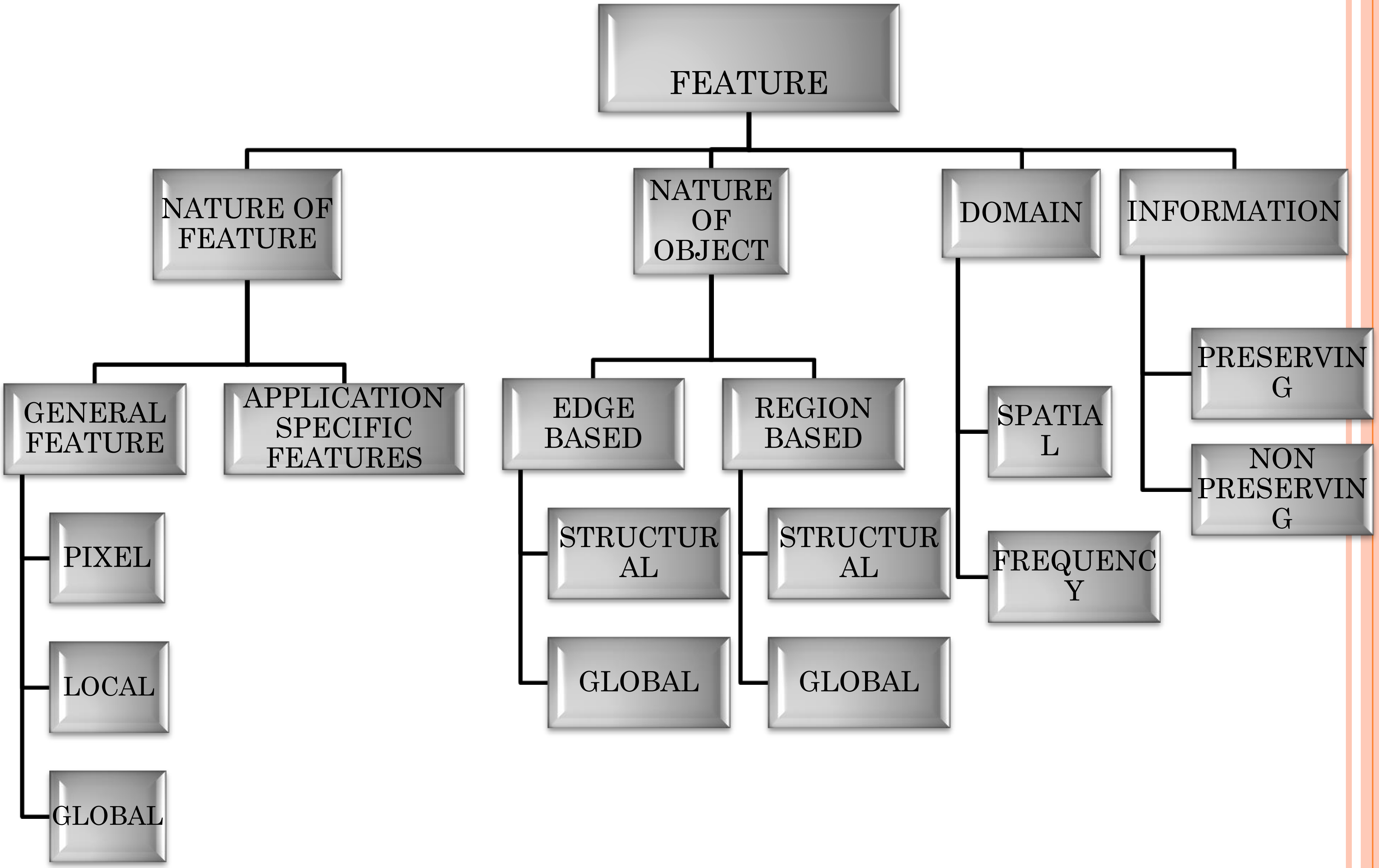


IMAGE

TO

OBJECT

CLASSIFICATION



FEATURE CLASSIFICATION

PROPERTIES:

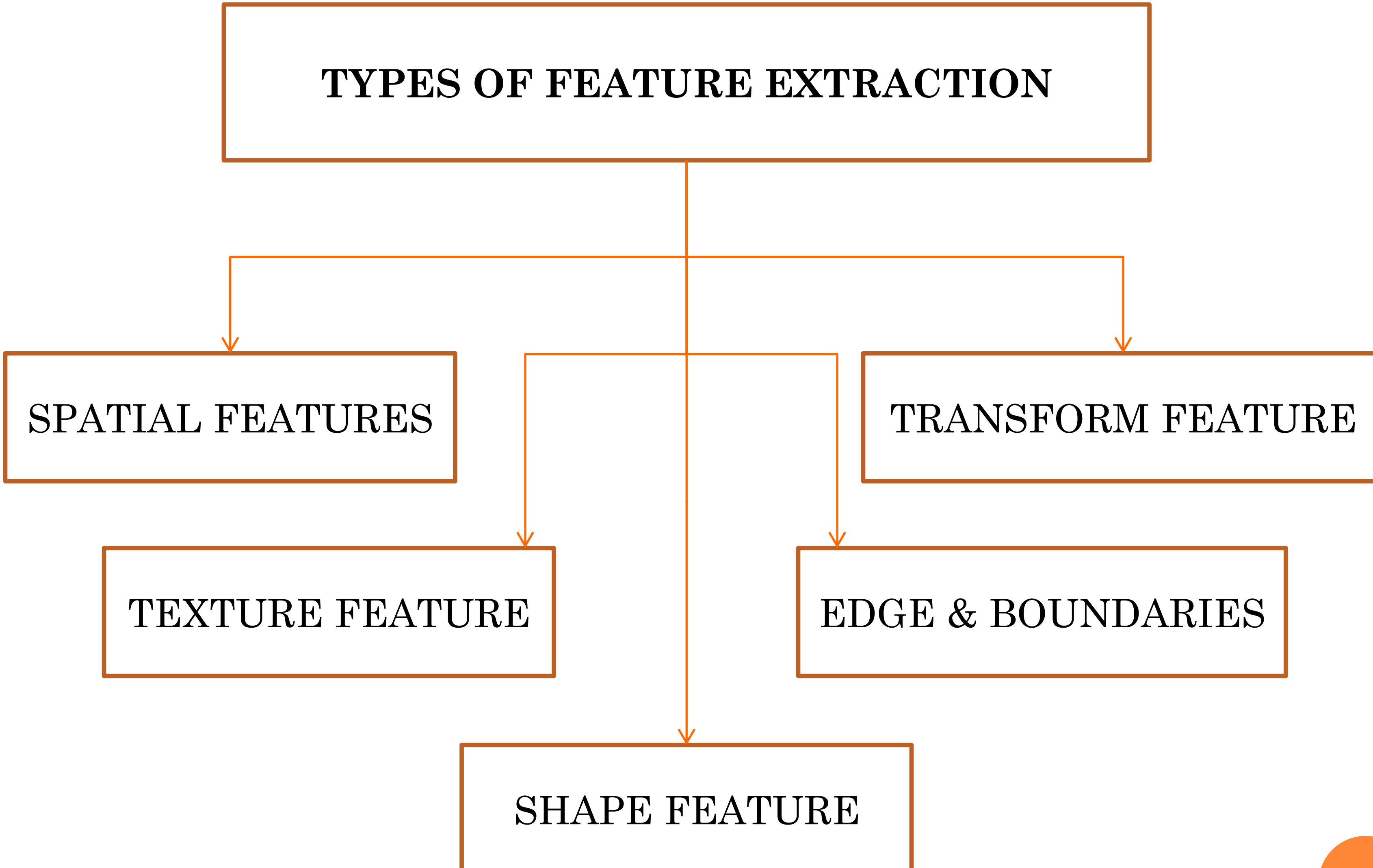
1) Instead of storing all the information about image, only object information is stored. It includes perimeter, area, compactness & center of mass etc.

2) The features should carry enough information about the image and should not require any domain-specific knowledge for their extraction.

3) There calculations should be easy so that approach will be feasible for a large image collection and rapid retrieval.

4) They should be related well with the human perceptual characteristics since users will finally determine the suitability of the retrieved images.

5) Features are mapped into a multi-dimensional feature space allowing similarity-based retrieval.


TYPES OF FEATURE EXTRACTION
SPATIAL FEATURES
TRANSFORM FEATURE
TEXTURE FEATURE
EDGE & BOUNDARIES
SHAPE FEATURE

Boundary Representation

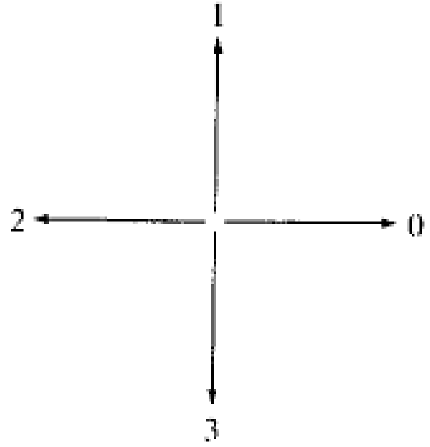
- After the processes of segmentation, object representation is done by joining the isolate points and forming boundary. This is required for proper image analysis.

- Some of the methods are:

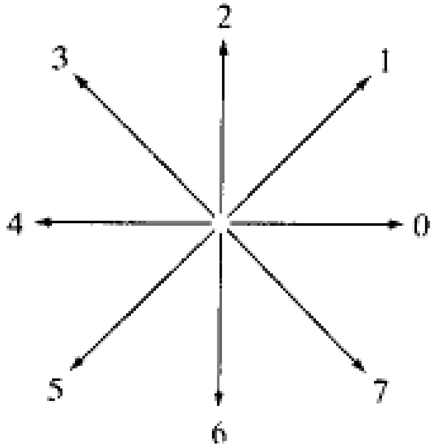
i. Chain code
ii. Polygonal approximations
iii. Signatures
iv. Bending energy

Chain code

- They are used to represent boundary by connecting lines having a specific length & orientation.



4-DIRECTIONAL CODE

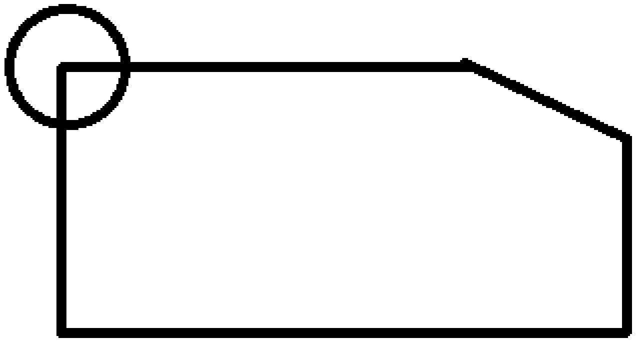


8-DIRECTIONAL CODE

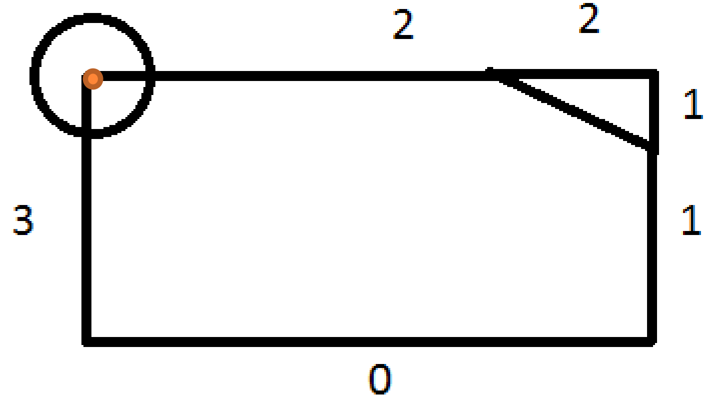
Algorithm for chain code

- From left to right start looking for object pixels.
- Start with object pixel 1, naming it as P_0.
- Set d=3, where d is direction variable.
- Label the pixel as current pixel & previous is labeled as previous pixel.
- Next pixel is obtained by searching the neighborhood in anticlockwise direction.

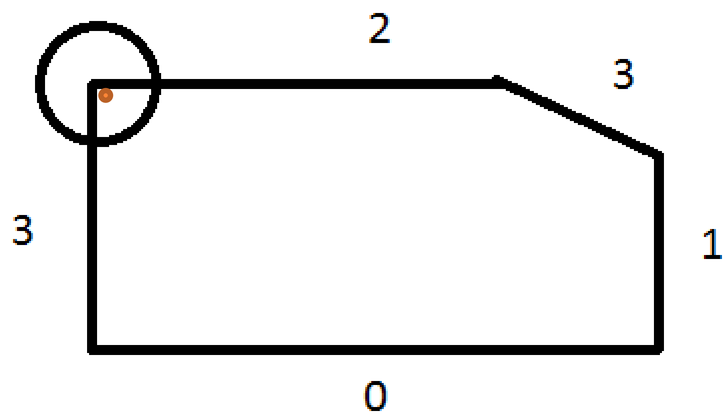
- Search neighborhood using (d+3)mod4.
- Now, label the current pixel as previous pixel & next pixel as current pixel.
- Update d.
- If the current boundary pixel P_0 is equal to $P_{n-1,}$ it means , if current pixel = previous pixel then stop. Else go back & repeat the again until starting pixel is reached.
- Label the region.
- Display $P_0 \ldots\ldots. P_{n-1}$ and exit.



ORIGINAL IMAGE



4-DIRECTIONAL CODE
301122



8-DIRECTIONAL CODE
301122

ADVANTAGE OF CHAIN CODE:

- Information of interest is preserved.
- It is a compact representation of an image.
- A chain code is prior condition for image analysis.
- It is translation invariant.
- But it is not rotation & scale invariant.
- It represents any change occurring in direction, indicating the corner points.

Problems in chain code:

1) Starting point of code determines chain code.

- So any change in rotation or size affects the chain code in unnormalized form.

2. Chain code is noise sensitive.

- To solve this problem , resampling of boundary is done to smooth out small variation.

Polygon approximation

- It is a process of connecting points of interest with approximation. Using fewest possible points this method is applied.
- Due to approximation loss of information occurs.
- The main reason to use this method is to approximately reduce the code for polygon.

There are 2 types of approximation:

1. *Fit & split*
2. *Merging*

1. Fit & split method

- This method is useful for following reasons:

i. Noise elimination
ii. Absence of relevant feature
iii. Absence of change in vertices
iv. Simplification of shape

- *Procedure for method :*

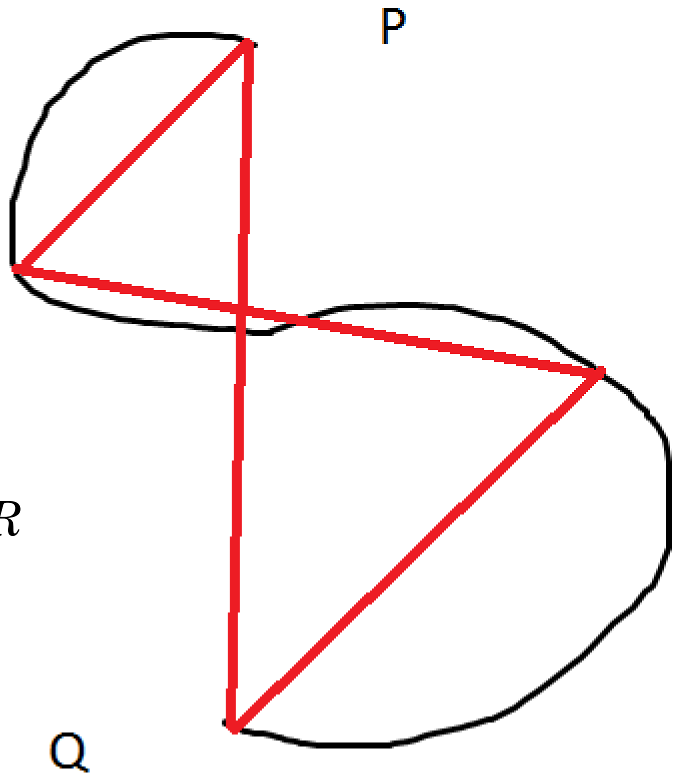
1. Divide boundary into small segment & fit a line for each segment.

2. Draw a line L b/w the 2 farthest points P & Q.

3. Compute the value of perpendicular line from boundary to line segment.

4. Compare this value with threshold value. If value exceeds threshold value then line is split.

5. Identify the vertices which is close to the inflection points(points where curve changes its sign) of curve.
6. Now connect all the points to obtain a polygon.
7. Repeat the above process until no split is further required.

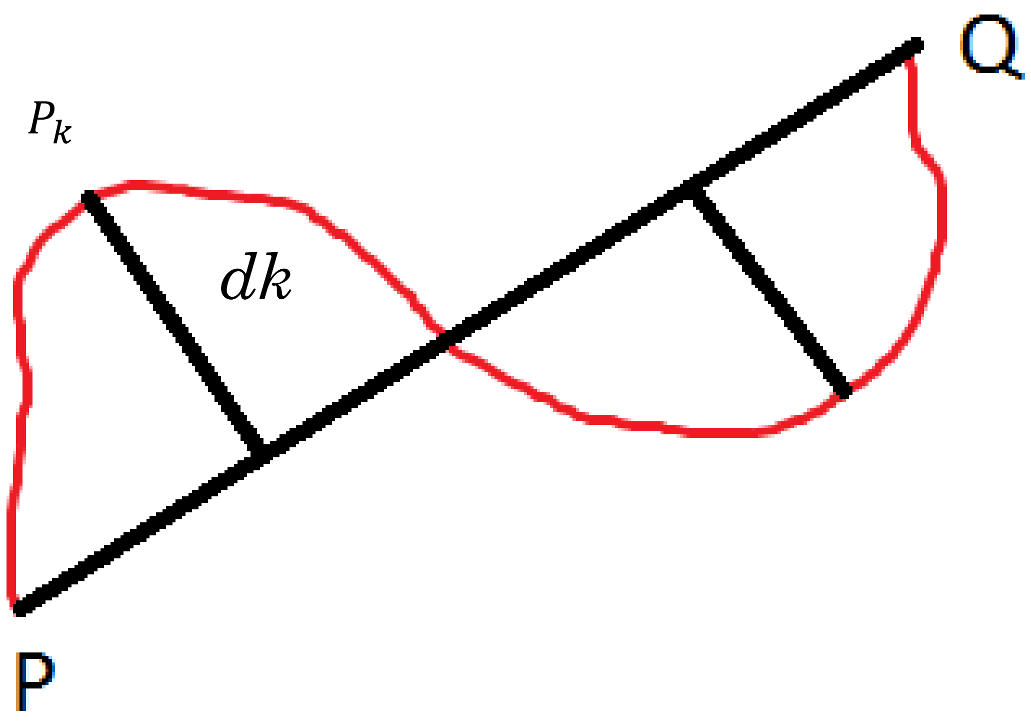


FIT & SPLIT PROCESS FOR S SHAPE IMAGE

Merging process

- Procedure for this method:

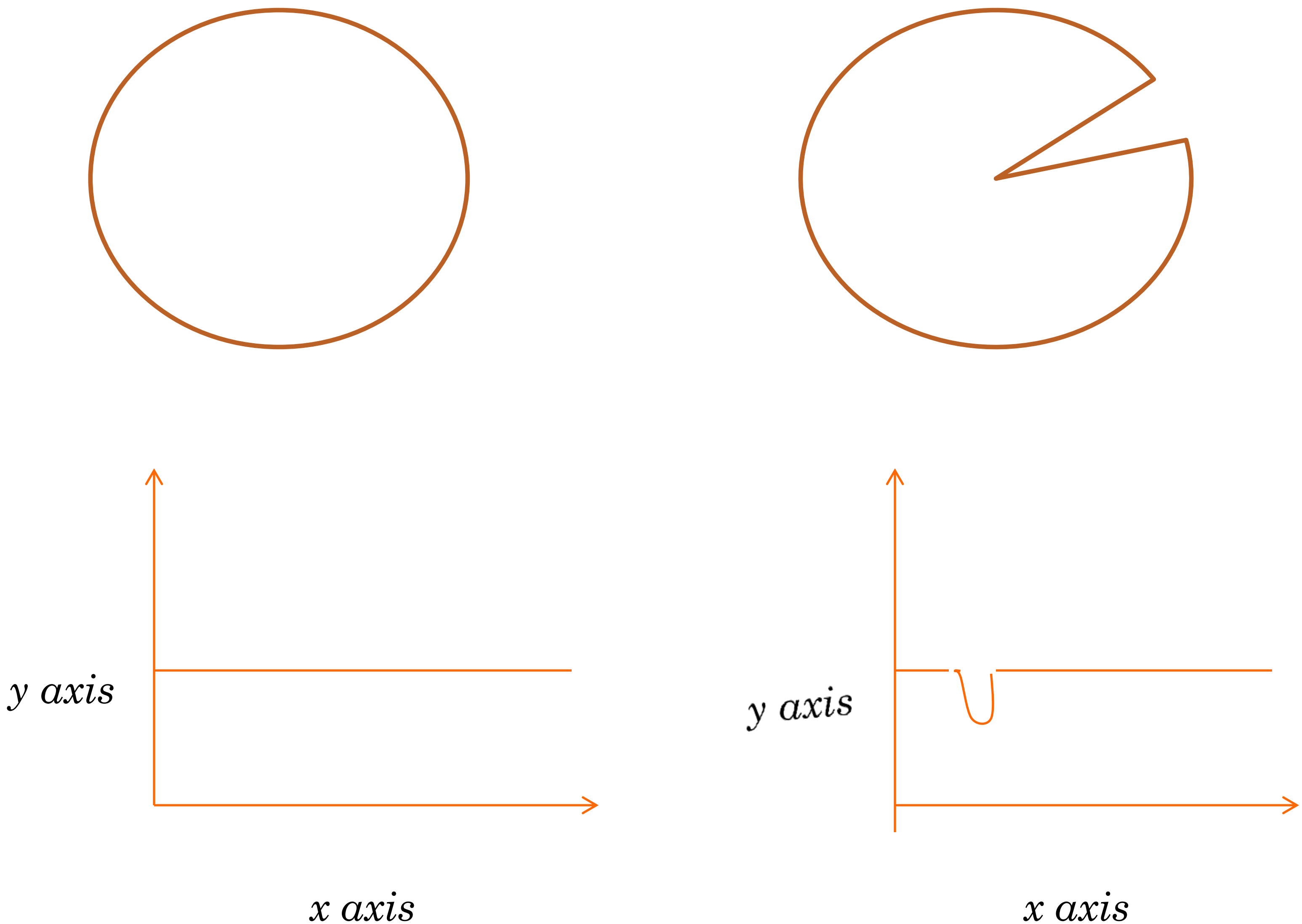
1. Select a starting point for of the curve.
2. Add pixels, one-by-one to line segment.
3. Let the line pass through the new pixels.
4. Compute the square error for all the points along the segment.
5. If error is greater than threshold, begin a new line at the current point.



MERGING PROCESS FOR S SHAPE IMAGE

Signatures

- Sometimes its required to represent 2D functions in 1D for fast & easy object matching. This is done using signature function.
- It is invariant to translation but not invariant of rotation and scaling.
- For e.g. x-axis shows the distance from centroid & y axis shows the angle. See example on next slide.


y axis
x axis
y axis
x axis

Bending energy

- Its another form of signature where $\varphi - s\ curve$ is used. $\varphi - s\ curve$ is a plot of tangent φ as a function of length around a curve.

- So this method store the value of tangent at each point of curve.

- It is useful because it is rotation invariant, translation invariant & in normalized form its scale invariant.

- Main problem with this method is that it doesn't allow reconstruction of original image.

- Equation:

$$\text{bending energy} = \frac{1}{L}\sum_{i=1}^{L} c^2(x)$$

Edge detection

Edge detection

- Edge detection of digital image is very important in the field of image analysis because the edge is boundary of the target and the background.

- **Only on obtaining the edge we can differentiate the target and the background.**

- It includes **image division, identification of objective region and pick-up of region shape, etc**.

Original image *Image after edge detection*

Example of edge detection

DEFINITION:

- Edge detection aims at identifying points in a digital image at which the image brightness changes sharply or has discontinuities.
- It is use to capture important events and changes in properties of the world.

- Discontinuities in image brightness can be represented by:

- discontinuities in depth,
- discontinuities in surface orientation,
- changes in material properties and variations in scene brightness.

APPROACH:

- The basic idea of image detection is to mark the partial edge of the image making use of edge enhancement operator firstly.
- Then we define the 'edge intensity' of pixels and extract the set of edge points through setting threshold.
- But due to existing noise & other flaws the borderline detected may produce wrong result.

SOLUTION OF PROBLEM:

Edge detection should contain the following two parts:

- Using edge operators the set of points of edge are extracted.
- Some of the edge points from the edge points set are removed and a number of edge points are filled in the edge points set.
- Then the obtained edge points are connected to be a line.

The common used edge enhancement operators are:

I. Differential operator

- *Roberts operator*
- *Sobel operator*
- *Prewitt operator*

II. Log operator

III. Canny operators

IV. Binary morphology, etc.

A. <u>DIFFERENTIAL OPERATOR:</u>

- **It can outstand grey change**.

- Some points gives large grey change. But on applying derivative operator it gives high value. So these differential values may be regarded as relevant 'edge intensity' and gather the points set of the edge by setting threshold value.

- First derivative is the simplest differential coefficient.

MATHMATICAL EQUATIONS :

- Suppose that the image is $f(x,y)$.
- Its derivative operator is the first order partial derivative $\partial f/\partial x$, $\partial f/\partial y$.
- They represent the rate-of-change that the gray f is in the direction of x and y.
- Equation for gray rate of change in α direction is:

$$\frac{\partial f}{\partial \alpha} = \frac{\partial f}{\partial x} \cos\alpha + \frac{\partial f}{\partial y} \sin\alpha$$

- The differential of the function is:

$$f = \frac{\partial f}{\partial x}\, dx + \frac{\partial f}{\partial y}\, dy.$$

- The direction derivative of function f(x,y) has a maxima at a certain point. And the direction of this point is:

$$\text{arc tan}\left[\left(\frac{\partial f}{\partial y}\right) / \left(\frac{\partial f}{\partial x}\right)\right].$$

- The maximum of direction derivative is:

$$\sqrt{\left(\frac{\partial f}{\partial x}\right)^2+\left(\frac{\partial f}{\partial y}\right)^2}$$

- The vector with this direction and modulus is called as the gradient of the function f, that is,

$$\nabla f(x, y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$$

So the gradient modulus operator is designed in the equation.

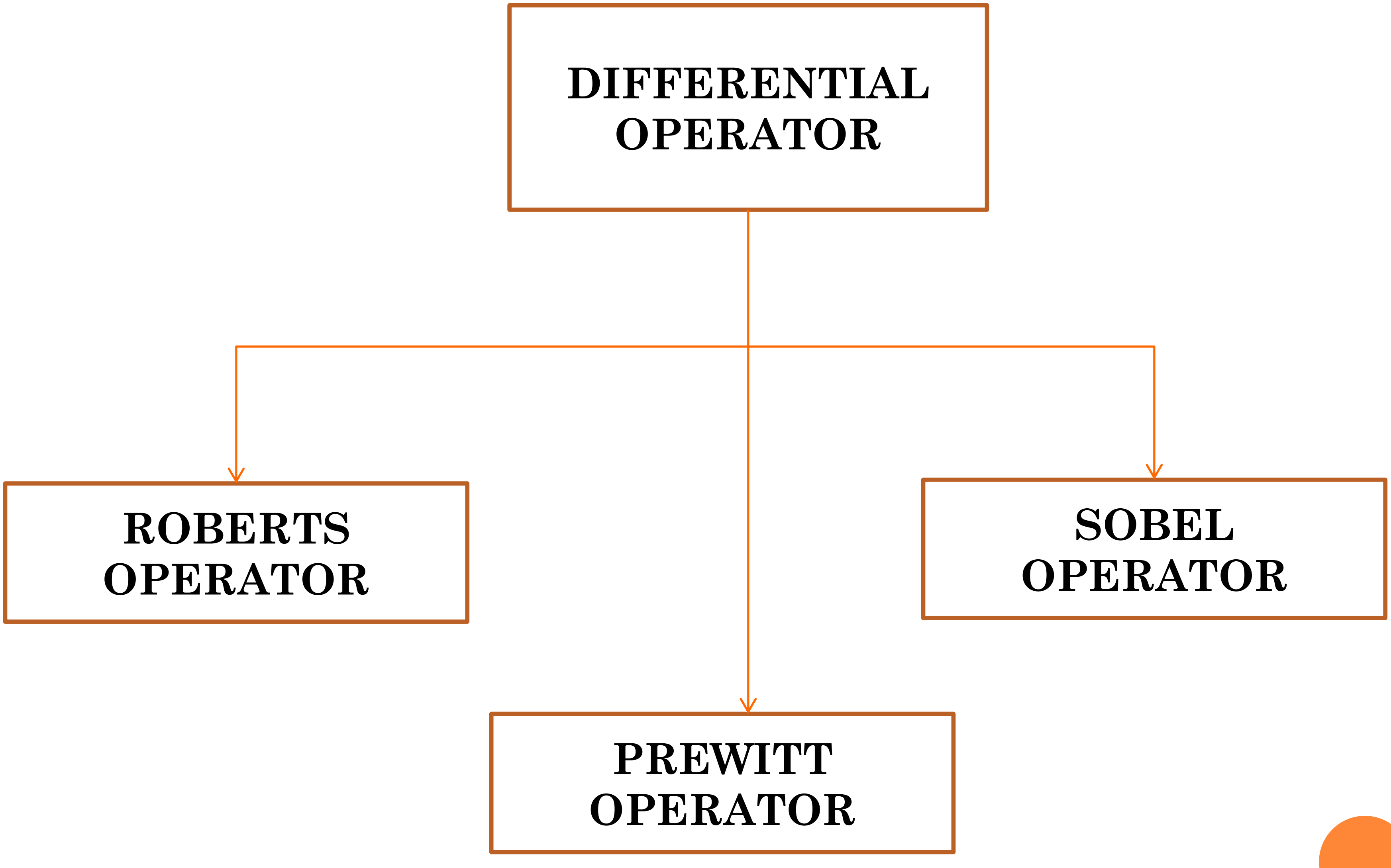
$$G[f(x,y)] = \sqrt{(\partial f / \partial x)^2 + (\partial f / \partial y)^2}$$

For the digital image, the gradient template operator is designed as:

$$G[f(i,j)] = [\Delta_x f(i,j)^2 + \Delta_y f(i,j)^2]^{1/2}$$

Where $\Delta_x f(i,j) = f(i,j) - f(i-1,j)$,

$$\Delta_y f(i,j) = f(i,j) - f(i,j-1) \ .$$


DIFFERENTIAL OPERATOR
ROBERTS OPERATOR
SOBEL OPERATOR
PREWITT OPERATOR

❖ Roberts operator

i. Roberts operator is a simple operator which uses concept of partial difference operator to outstand edge.
ii. It is best suited for image with steep low noise.
iii. The edge location is not very accurate by Roberts operator as borderline obtained is quite thick of extracted image.
iv. Equation for Roberts Operator:

$$G[f(x,y)] = \{[f(x+1,y+1) - f(x,y)]^2 + [f(x+1,y) - f(x,y+1)]^2\}^{1/2}$$

❖ <u>Sobel & Prewitt operator:</u>

i. Sobel operator works same as Prewitt operator but its edge detection is wider the Sobel and

ii. Prewitt operators have a better effect for images with grey level changing gradually and more noises.

i. *Prewitt operator*:

- To reduce the influence of noise when detecting edge, it enlarges edge detection operator template from 2 X 2 to 3 X 3 to compute difference operator.
- It can not only detect edge points, but also restrain the noise.

ii. *Sobel operator*:

- It counts difference using weighted for 4 neighborhoods on the basis of the Prewitt operator.

❑ **Prewitt operators calculation**:

Suppose that the pixel number in the 3 × 3 sub-domain of image is as follows:

$$\begin{matrix} A_0 & A_1 & A_2 \\ A_7 & f(i,j) & A_3 \\ A_6 & A_5 & A_4 \end{matrix}$$

We order that

$$X = (A_0 + A_1 + A_2) - (A_6 + A_5 + A_4)$$

&

$$Y = (A_0 + A_7 + A_6) - (A_2 + A_3 + A_4)$$

Then Prewitt operator is as follows:

$$G[f(i,j)] = (X^2 + Y^2)^{1/2}$$

Or

$$G[f(i,j)] = |X| + |Y|$$

Prewitt operator in the form of the template:

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

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

Prewitt operator

Original image

Prewitt edge magnitude

❑ **Sobel operator's calculation**:

Sobel operator can process those images with lots of noises and gray gradient well. We order that

&

$$X = (A_0 + 2A_1 + A_2) - (A_6 + 2A_5 + A_4)$$

$$Y = (A_0 + 2A_7 + A_6) - (A_2 + 2A_3 + A_4)$$

Then Sobel operator is as follows:

$$G[f(i,j)] = (X^2 + Y^2)^{1/2}$$

Or

$$G[f(i,j)] = |X| + |Y|$$

The template of the Sobel operator is shown below:

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

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

Sobel operator

Original image

Sobel edge magnitude

B. LOG OPERATOR:

- It is a **linear** and **time-invariant operator**.
- It detects edge points through searching for spots with two-order differential coefficient is zero in the image grey levels.
- The operator is very sensitive to noise.
- For a continuous function *f(x,y)* , the Log operator is defined as at point *(x, y)* :

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

- The Log operator is the process of filtering and counting differential coefficient for the image.
- It determines the zero overlapping position of filter output using convolution of revolving symmetrical Log template and the image.
- The Log operator's template is shown:

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

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

Log operator

Original image

Log edge magnitude

Detection process of the Log operator:

a. Firstly pre-smooth the image with Gauss low-pass filter.

b. Then using the Log operator find the steep edge in the image.

c. Then we perform Binarization with zero grey level to obtain the closed, connected outline and eliminate all internal spots.

C. **<u>CANNY OPERATOR</u>**:

- Canny extracts thin, clear edges & better method for edge detection .
- This method is not easily disturbed by noise and can keep the good balance between noise and edge detection.
- It can detect the true weak edge.
- It works by a two step algorithm:

i. Firstly, an edge filter just like Sobel is applied.

ii. Secondly, to obtain thin lines of edges, a non-maxima suppression is applied.

Calculation of Canny Operator

1. The first step is done by using sobel filter.
2. The second step is important for getting thin edges.

a. The Canny operator uses two thresholds to detect strong edge and weak edge respectively.

b. Output value will contain weak edge only when a strong edge is connected to it.

c. It searches for the partial maximum value of image gradient which is counted by the derivative of Gauss filter.

d. The theory basis of canny operator is shown below:

Gauss: $$g(x,y) = \exp[-(x^2 + y^2)/2\sigma^2]$$

Edge normals: $$n_\perp = \nabla(g * P) / \left|\nabla(g * P)\right|$$

Edge strengths: $$G_n P = \frac{\partial}{\partial n_\perp}[g * P]$$

Maximal strengths: $$0 = \frac{\partial}{\partial n_\perp} G_n P = \frac{\partial^2}{\partial {n_\perp}^2}[g * P]$$

Where σ standard deviation

P is threshold

Original image

Canny Edge extraction

D. <u>BINARY MORPHOLOGY:</u>

- It is a better approach then differential treatment as it is not sensitive to noise, and the edge extracted is relatively smooth.
- Binary image is also known as black-and-white image. So object can be easily identified from the image background.
- Here we adopt the combination of binary image and mathematical morphology to detect edge.

Procedure of Binary Morphology

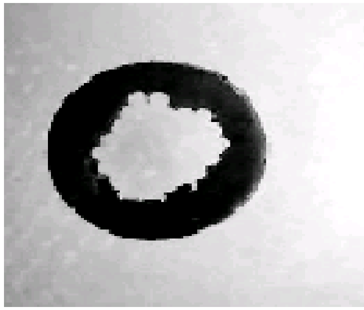
- Suppose that the region is shown in form of the set A. Its border is ϐ(*A*) . B is an appropriate structure element, and it is symmetrical around the origin.
- Firstly we corrupt A with B recorded as:

$$A \Theta B = \{x \mid (B)_x \subseteq A\},$$

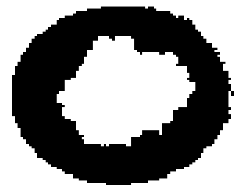
where $(B)_x$ is a translation along the vector.

- The interior of region is available with $A\Theta B$.
- And $A-(A\Theta B)$ is the borderline naturally.
- Then $\beta(A)$ is obtained.
- The equation of edge extraction can be said $\beta(A) = A-(A\Theta B)$.
- Structuring element is larger, the edge gained will be wider

Comparison of several edge detection algorithms

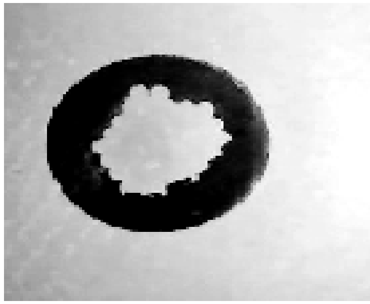


Original image

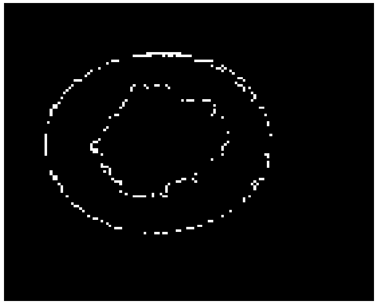


Binary image

Edge extraction

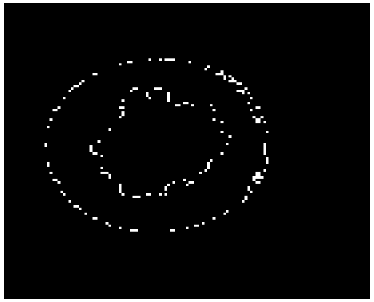


Original image



Robert operator

Sobel operator



Prewitt operator

Canny operator

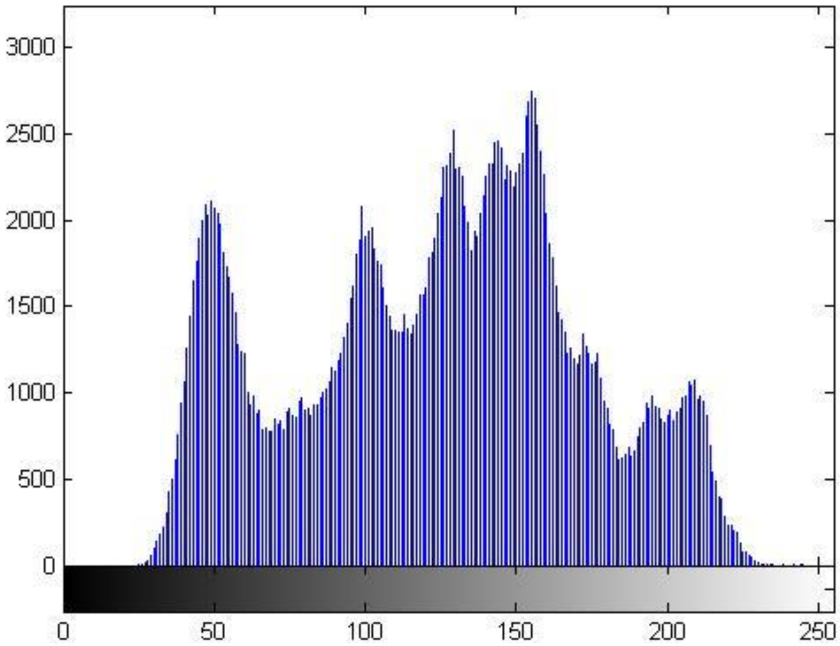
Log operator

Spatial feature

Spatial feature

- It is simplest & most important spatial feature of an object. This feature explains the various physical quantities such as reflectivity, transitivity etc.
- It is based on luminance or intensity feature.
- Also known as amplitude or histogram or brightness feature.
- The brightness of an image can convey many useful information about the image.

grayscale image



histogram of grayscale image

- Some of the feature described as spatial features are:

1. Mean
2. Standard deviation
3. Skewness
4. Kurtosis
5. Energy
6. Entropy
7. Autocorrelation
8. Inertia
9. Absolute value , etc.

- Mean: it indicates brightness of image

$$b' = \sum_{b=0}^{L-1} b \cdot p(b)$$

Where $p(b) = \frac{N(b)}{m}$, $N(b)$ is the number of pixels having amplitude r & gray levels L. m is the number of pixels in image.

- Standard deviation: it indicates contrast of image.

$$\sigma = \sqrt{\sum_{b=0}^{L-1} [(b - b')^2 p(b)]^{\frac{1}{2}}}$$

- Skewness: it indicates asymmetry about mean.

$$=\frac{1}{\sigma^3}\sum_{b=0}^{L-1}(b-b')^3 p(b)$$

- Kurtosis: it refers to the weight of distribution toward tail.

$$=\frac{1}{\sigma^4}\sum_{b=0}^{L-1}(b-b')^4 p(b)\text{-}3$$

- Energy: it is sum of brightness values of all the pixels present in an object.

$$=\sum_{b=0}^{L-1}[p(b)]^2$$

- Entropy: it indicate minimum number of bits required to code image.

$$= - \sum_{b=0}^{L-1} p(b) \log_2 p(b)$$

Measure of energy along diagonal:

- <u>Autocorrelation</u>: finds the repeating pattern in an image.

$$S_A = \sum_{a=0}^{l-1} \sum_{b=0}^{l-1} a.b.p(a,b)$$

Where $p(a,b) = \frac{N(a,b)}{m}$, *p(a,b) is joint probability distribution of pixels a & b. m* is the number of pixels in image.

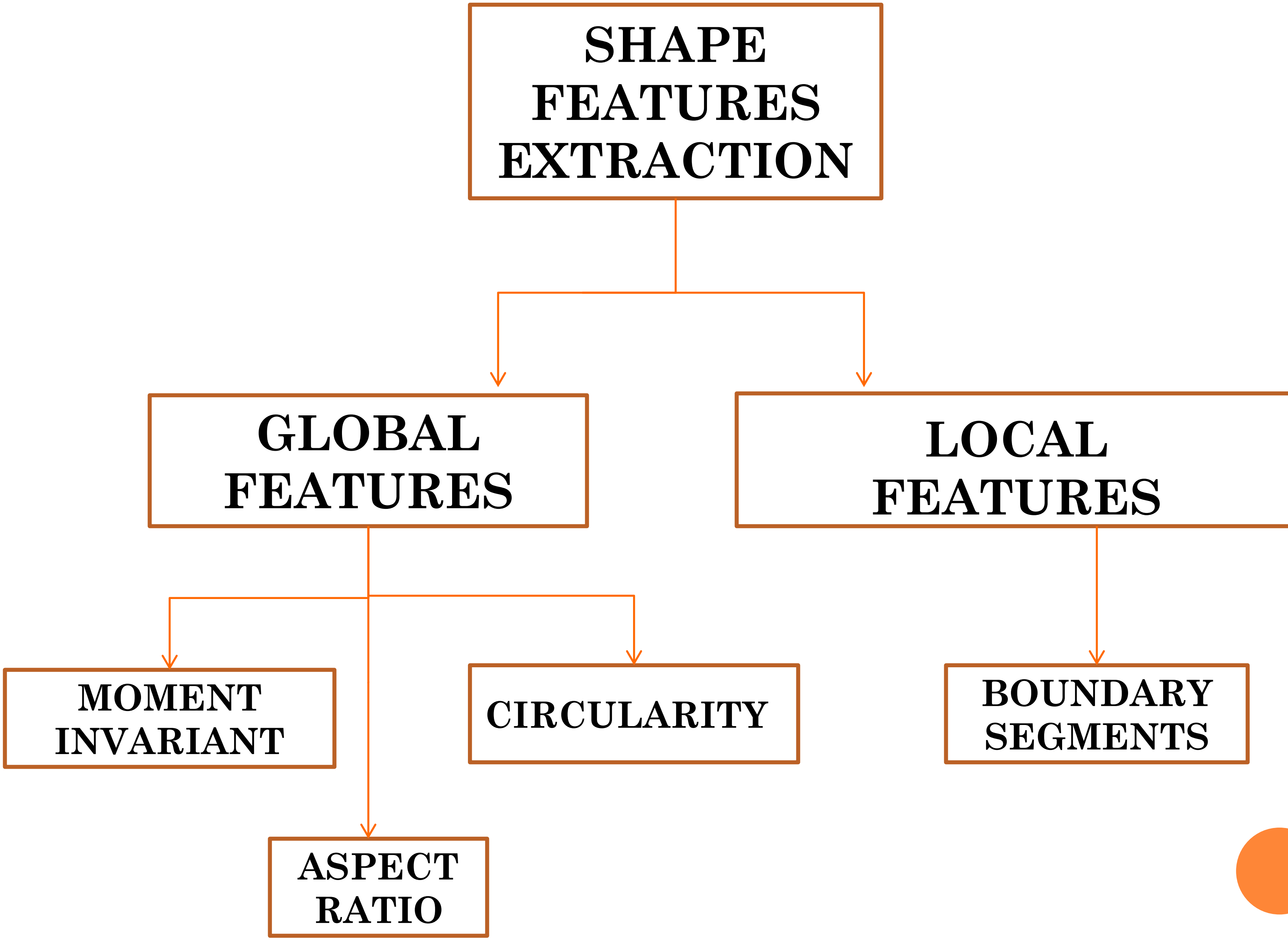
- Inertia :

$$S_I = \sum_{a=0}^{l-1} \sum_{b=0}^{l-1} (a-b)^2 b.\,p(a,b)$$

- Absolute value: it subtracts each element in b from corresponding value of a.

$$S_v = \sum_{a=0}^{l-1} \sum_{b=0}^{l-1} |a-b|.\,p(a,b)$$

Geometric feature


SHAPE FEATURES EXTRACTION
GLOBAL FEATURES
LOCAL FEATURES
MOMENT INVARIANT
CIRCULARITY
BOUNDARY SEGMENTS
ASPECT RATIO

- Shape is characterized by boundaries , giving visual appearance to an object. It is independent of orientation & position of an object.

- Analysis is done based on two categories:
  - 2D shapes :It informs about boundary (thin shape) & region(thick shapes)
  - 3D shapes : It informs about involves about depth)

COMMON SHAPE FEATURES

AREA

- the area of an object is the number of pixels contained in the object.
- It is shift invariant. It means it remains constant on applying shift operation
- But it is size variant. It means that area changes with size.
- It includes the area inside the boundary, including boundary.
- Doesn’t include hole or gap inside object.
- *Area of binary image:*

$$A=\sum_{i=1}^{n}\sum_{j=1}^{m}B(i,j)$$

PERIMETER

- the perimeter of an object is the number of pixels present on boundary of an object.
- Perimeter is shift invariant, rotation invariant (independent of rotation) but not size invariant.
- It can be calculated by using chain code as follows:

i. For a 4-directional code:

P=number of chain codes

ii. For an 8-directional code:

$$P = \text{even count} + \sqrt{2}(odd\ count)\text{units.}$$

SHAPE FACTOR(COMPACTNESS)

$= Perimeter^2 \,/\, area$

AREA TO PERIMETER RATIO

- It represents the roundness, circularity, thinness ratio.
- It a dimensionless quantity.
- It value varies from 0 to 1.
- Value 1 represents a perfect circle.
- $\frac{A}{P} = 4\,\pi\, x\, \frac{area}{perimeter^2}$

MAJOR AXIS(OBJECT LENGTH)

- The longest line that can be drawn connecting the 2 farthest points of an object.
- If $(x_1,\ y_1)$ & $(x_2,\ y_2)$ are the two points, then :

i. the length of longest line can be calculated by:

$$\sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)}$$

ii. orientation can be calculated by:

$$\tan^{-1} \frac{y_2 - y_1}{x_2 - x_1}$$

MINOR AXIS(OBJECT WIDTH)

- The longest line that can be drawn perpendicular to major axis.

- If (x_1, y_1) & (x_2, y_2) are the two points, then the length of longest line can be calculated by:

$$\sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)}$$

BOUNDARY BOX AREA

- It is the smallest rectangle that can contain the whole object.

- It is of 2 types:

i. Feret box
ii. Minimum bounding rectangle.

- **Boundary box calculation:**

i. Boundary box area= major axis X minor axis

ii. Elongatedness

$$= \text{length of major axis/perimeter}$$

iii. Aspect ratio

$$= \frac{\text{length of the region bounding rectangle}}{\text{width of the region bounding rectangle}}$$

iv. Rectangularity

$$= \frac{\text{region area}}{\text{area of the minimum bounding rectangle}}$$

SPATIAL MOMENTS

GREY CENTROID: it is a point of image having equal brightness in all direction.

G(x,y) calculation:

$$x=\frac{\text{Sum of (pixel x co-ordinates X pixel brightness)}}{\text{Sum of pixel brightness of the object}}$$

$$y=\frac{\text{Sum of (pixel y co-ordinates X pixel brightness)}}{\text{Sum of pixel brightness of the object}}$$

CENTER OF MASS:

- Center of mass x

$$= \frac{\text{sum of the object's x}-\text{pixel co}-\text{ordinates}}{\text{the number of pixels in the object}}$$

- Center of mass y

$$= \frac{\text{sum of the object's y}-\text{pixel co}-\text{ordinates}}{\text{the number of pixels in the object}}$$

CENTRAL MOMENTS

$$\mu_{ij}=\sum_x \sum_y (x - x')^i (y - y')^j f(x,y)$$

Normalized central moments:

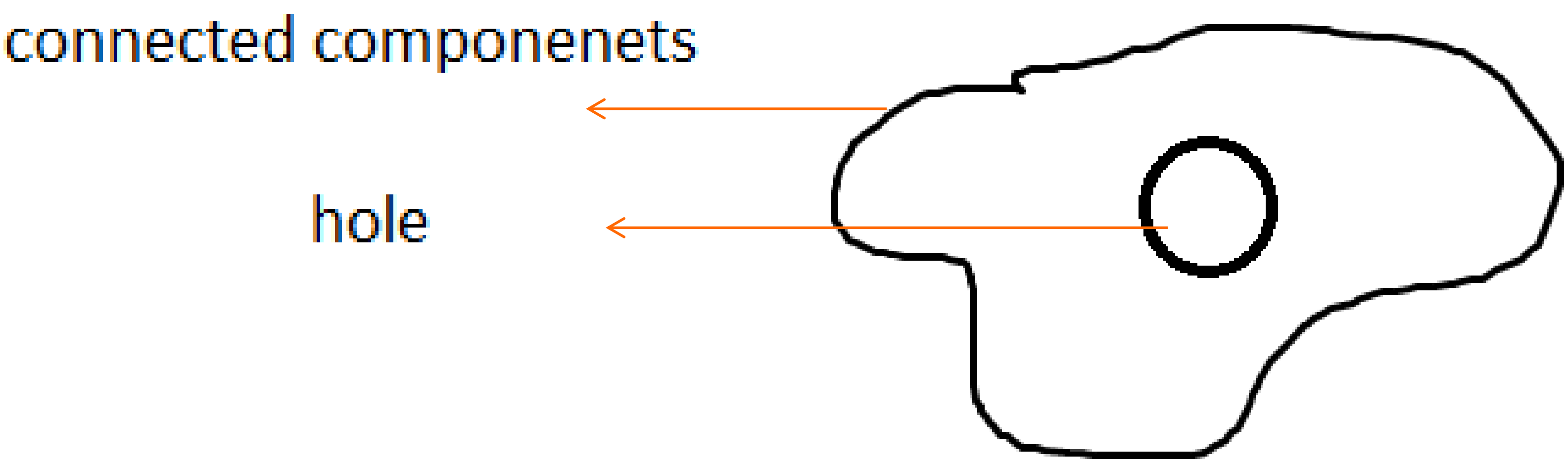
$$\eta_{ij} = \frac{\mu_{ij}}{{\mu_{00}}^{\lambda}}$$

Where $\lambda = \left(\frac{i+j}{2}\right) + 1 \; and \; (i + j) \geq 2$

Topological features

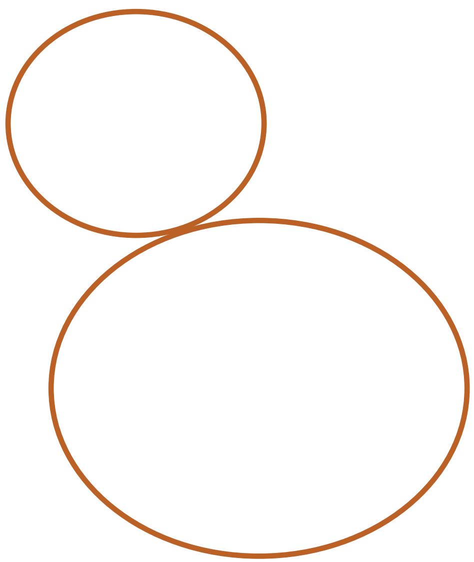
- Useful for recognition of object & is better because not affected by deformation.

1. <u>Hole:</u> transformation do not affect presence of hole.



2. <u>Connect component</u>: area populated by pixels having common properties.
3. <u>Euler number</u>:

E=Connected component-Holes



euler number=1-2

=-1

4. Number of holes present: number of holes present inside an object.

5. Total area of holes: total area inside hole present in pixels.

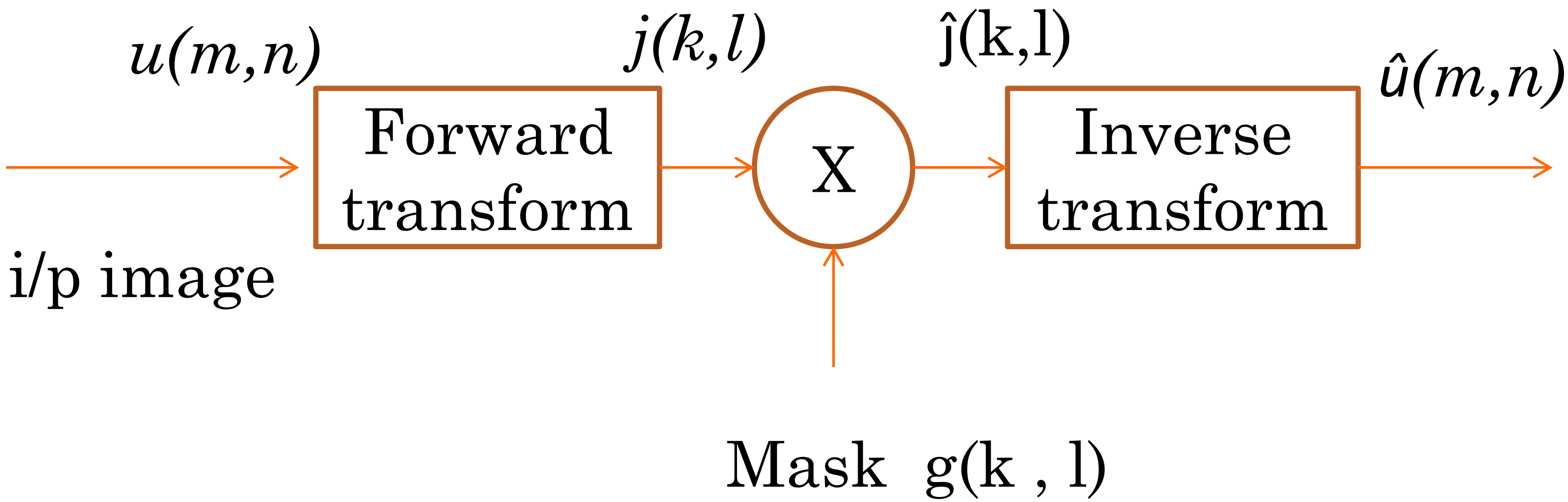
6. Total object area: its value ranges from 0 to 1. it measure the object proliferation. Value 1 represents that whole object is a hole.

7. Number of connected components

Transform feature

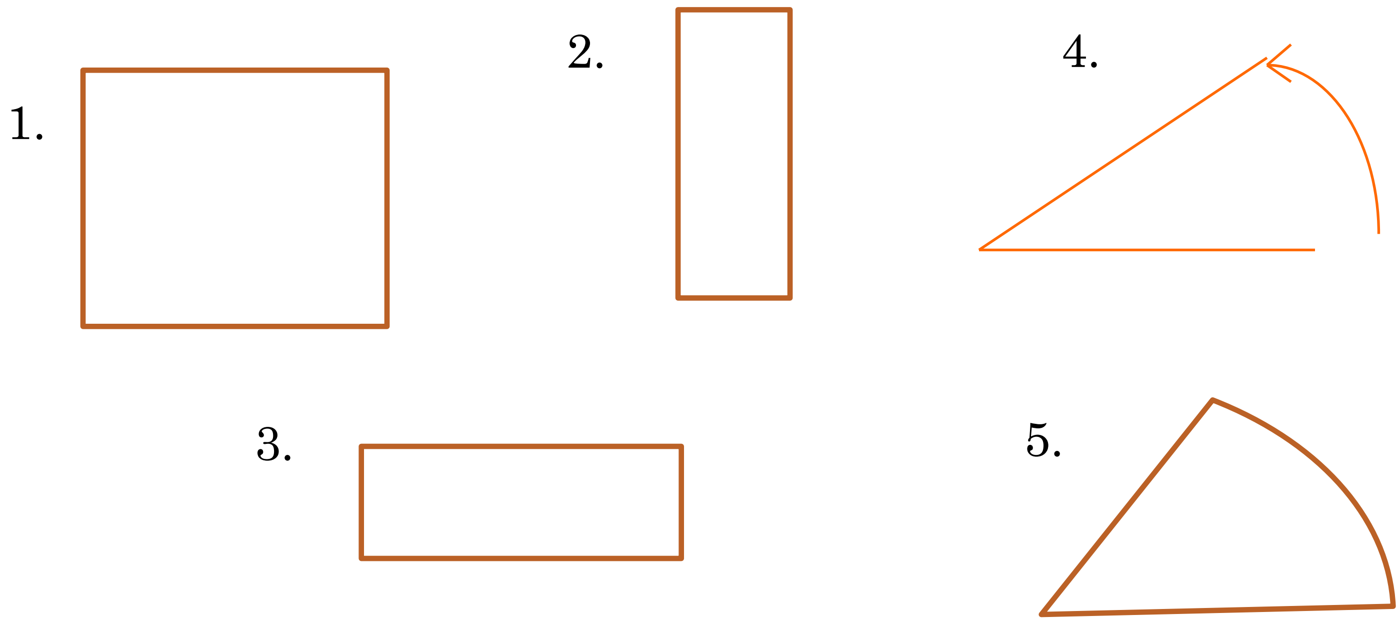
TRANSFORM FEATURES

- It provides information of data in frequency domain.
- It is used to examine specific region of an image feature using mask.
- Transform feature can be extracted using zonal filter or mask.
- Generally high frequency is used for edge & boundary detection and for orientation detection angular silts can be used.


u(m,n)
i/p image
Forward transform
j(k,l)
X
ĵ(k,l)
Inverse transform
û(m,n)
Mask g(k , l)

Some of the useful mask are:

1. BOX
2. VERTICAL SLIT
3. HORIZONTAL SLIT
4. RING
5. SECTOR



Some of the feature obtained from spectral component are:

I. POWER: $|F(u,v)|^2$

II. SPECTRAL REGION POWER: a portion of spectral element is taken & its spectral region power is calculated using formula:

for box region:

$$=\sum_{u=0}^{N-1} \sum_{v=0}^{N-1} |F(u,v)|^2$$

for ring region: it is defined by 2 radii, r_1 and r_2

$$r_1 \leq u \leq r_2$$

$$= \pm\sqrt{r1^2 - u^2} \leq v \leq \pm\sqrt{r2^2 - u^2}$$

for sector region:

$$\theta_1 \leq \tan^{-1}\frac{v}{u} < \theta_2$$

$$u^2 + v^2 = r^2$$

FOURIER DESCRIPTORS

- For any sampled boundary :

$$u(n) \triangleq x(n) + jy(n),$$

where n=0,1,....,N-1,

x(n)& y(n) are pair of waveform.

- For closed boundary u(n) will be periodic with period N.

- DFT representation for closed boundary :

1. $$u(n) \triangleq \frac{1}{N}\sum_{k=0}^{N-1} a(k)\exp(\frac{j2\pi kn}{N}),$$

   $0 \leq n \leq N-1$

2. $$a(k) \triangleq \frac{1}{N}\sum_{k=0}^{N-1} u(n)\exp(\frac{-j2\pi kn}{N}),$$

   $0 \leq k \leq N-1$

- a(k) is known as Fourier descriptors(FDs) of boundary.

- For continuous boundary function.

$$u(n) \triangleq x(n) + jy(n),$$

where n=0,1,....,N-1

x(n)& y(n) are pair of waveform.

- But FDs are its Fourier series coefficient & is infinite.