Questions and Answers

Region Identification

Q1. Explain the algorithm for region identification

Or Describe the procedure of region identification using 4N and 8N

Contour-based shape representation and description

Q2. Describe contour based shape representation with respect to

i) Chain code ii) Border representation

Q3. Briefly explain any three border representation techniques

Or Discuss any three contour based shape representation techniques with example.

Region-based shape representation and description

Q4. Explain region based shape representation and description

Q5. Discuss in detail the different scalar region descriptors.

Or Explain any three region based shape representation models.

Q6. What are the factors need to be considered for a robust shape representation? Discuss any two region based approaches which address these factors

Q7. Explain region decomposition with a neat diagram.

Q8. Explain region neighborhood graphs

Shape Classes

Q9. What are shape classes? Suggest some approaches to identify the shape class of an object in an image.

Other Questions

- Write a note on Shape invariants.
- Write an algorithm to find region area from 4-connectivity chain code
- Explain Convex Hull based region representation
- Give the method to find skeleton by thinning

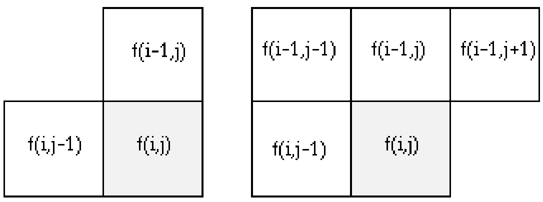
Q1. Explain the algorithm for region identification

Or Describe the procedure of region identification using 4N and 8N

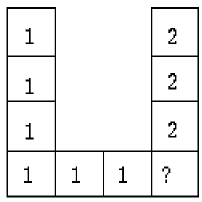
Region identification is necessary for region description. One method for region identification is to label each region (or each boundary) with a unique (integer) number. This process is called as labelling or coloring or connected component labelling.

Algorithm: Region identification using 4N and 8N
Input: Image f, with foreground pixels f(x, y)=1 and with background pixels f(x, y)=0
Output: Labeled image f(x, y)
Method:

1. **First pass:** Scan the image row by row. For each foreground pixel (f(x,y)=1) assign a label looking into its neighbours. Here neighbours are the pixels that are already scanned and in 4N or 8N connectivity (Refer the figure given below). For a pixel f(i,j) the label is assigned as follows
   - If all the neighbours are background pixels, f(i,j) is assigned with a new unused label.
   - If there is just one neighbouring pixel with a non-zero label , assign this label to the pixel f(i,j) .
   - If there is more than one neighbouring pixel with a non-zero label, assign the label of any one of the pixel to f(i,j). If the labels of any two neighbours differ (label collision), store the label pair as being equivalent in an equivalence table.
2. **Second pass**: All of the region-pixels were labeled during the first pass but some regions have pixels with different labels (due to label collisions). The whole image is scanned again, and pixels re-labeled using the equivalence table information

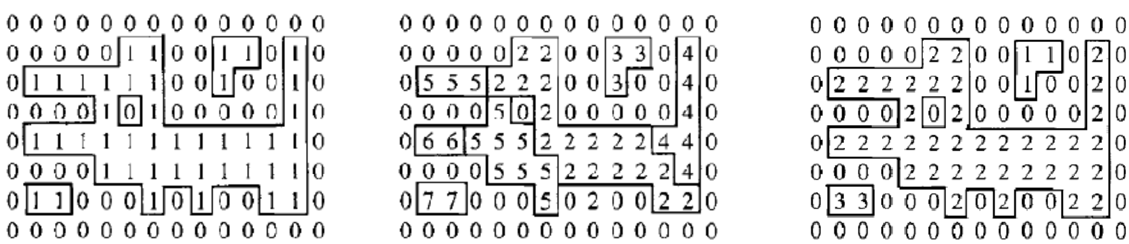


Neighbours of f(i,j) in 4N and 8N



Label Collision

Illustration



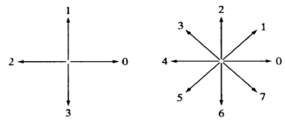
(a) (b) (c)

Region identification in 8-connectivity. (a), (b), (c) Algorithm steps.
Equivalence table after step (b): **2-5\, 5-6, 2-4.**

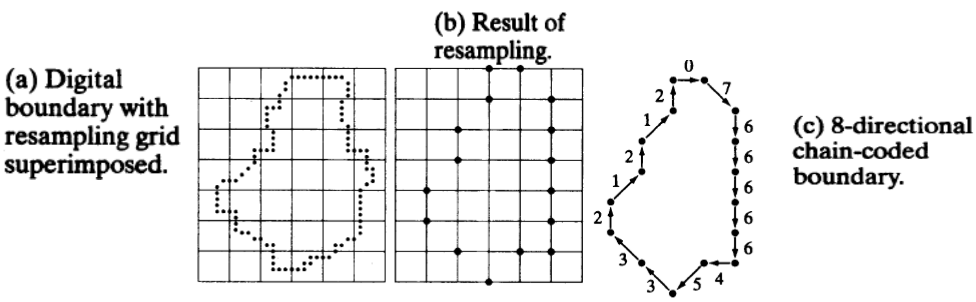
Q2. Describe contour based shape representation with respect to i) Chain code ii) Border representation

Chain codes

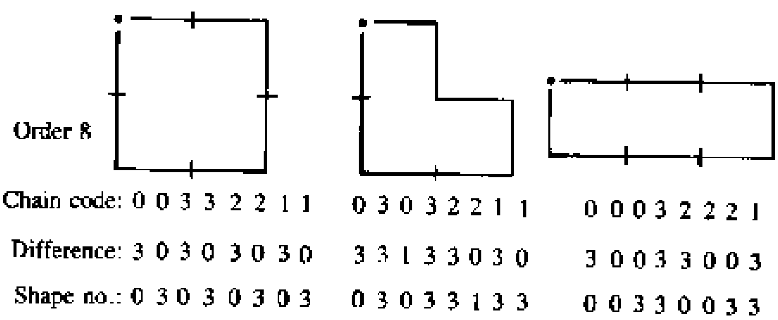
- Chain codes describe an object by a sequence of unit-size line segments with a given orientation.



4-directional and 8-directional chain codes



- The first element of such a sequence must bear information about its position to permit the region to be reconstructed. The process results in a sequence of numbers.
- If the chain code is used for matching it must be independent of the choice of the first border pixel in the sequence.
- One possibility for normalizing the chain code is to find the pixel in the border sequence which results in the minimum integer number. This makes the representation starting point independent.
- We can make the chain code rotation independent by taking its derivative. A mod 4 or mod 8 differences is called as chain code derivative. It is another numbered sequence that represents relative directions of region boundary elements, measured as multiples of counter clockwise 90 degree or 45 degree direction changes.

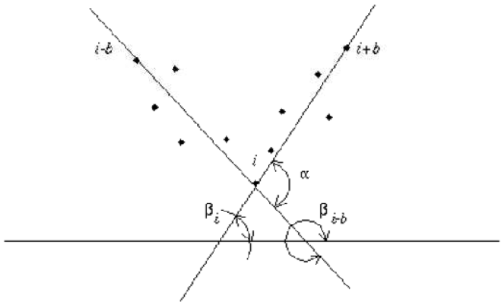


Border representation

The following descriptors are mostly based on geometric properties of described regions

Boundary length: It is the total number of pixels in the boundary

Curvature: Curvature is defined as the rate of change of slope. Values of the curvature at all boundary pixels can be represented by a histogram; relative numbers then provide information on how common specific boundary direction changes are. Histogram of boundary angles, such as β angle in figure, can be built in a similar way - such histograms can be used for region description.



Bending energy: It is the Energy needed to bend the rod to the desired shape. It is computed as the sum of squares of curvature

$$BE = \frac{1}{L}\sum_{k=1}^{L} c^2(k)$$

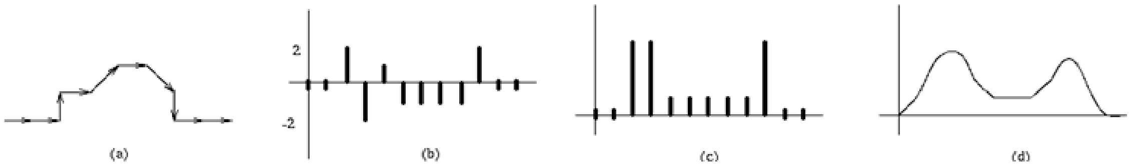


Figure *Bending energy: (a) Chain code 0, 0, 2, 0, 1, 0, 7, 6, 0, 0, (b) curvature 0, 2, -2, 1, -1, -1, -1, 2, 0, (c) sum of squares gives the bending energy, (d) smoothed version.*

Signature: It is the sequence of normal contour sequences. Signatures can be used for the recognition of overlapping objects also. But this is sensitive to noise.

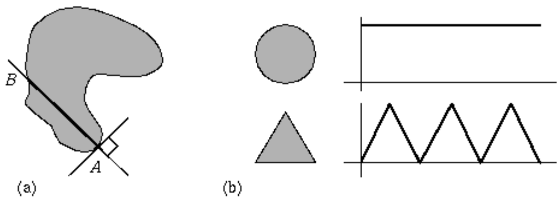


Figure 6.9 *Signature: (a) Construction, (b) signatures for a circle and a triangle.*

Chord: Chord is a line joining any two points of the region boundary. It is the distribution of the lengths and angles of all chords on a contour is used for shape description. It is the total count of chords having a fixed length and angle. Let b(x,y)=1 represent the contour points, and b(x,y)=0 represent all other points.

The chord distribution can be computed as

$$h(\Delta x, \Delta y) = \sum_i \sum_j b(i,j) b(i+\Delta x, j+\Delta y)$$

Rotation-independent radial distribution:

$$h_r(r) = \int_{-\pi/2}^{\pi/2} h(\Delta x, \Delta y) r d\theta$$

The angular distribution h(θ) is independent of scale, while rotation causes a proportional offset

$$h_a(\theta) = \int_0^{\max(r)} h(\Delta x, \Delta y) dr$$

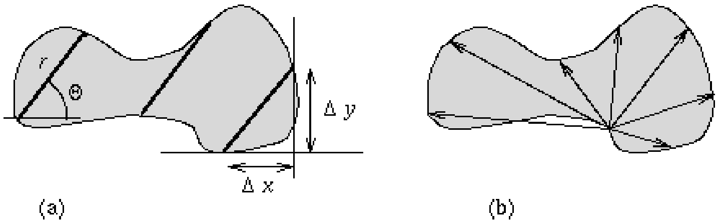


Figure 6.10 *Chord distribution.*

Q3. Briefly explain any three border representation techniques
Or Discuss any 2 contour based shape representation techniques with example.

Fourier transforms of boundaries

- Suppose C is a closed curve (boundary) in the complex plane. Traveling anti-clockwise along this curve keeping constant speed, a complex function z(t) is obtained, where t is a time variable.

$$z(t) = \sum_n T_n e^{int}$$

- The Fourier descriptors can be invariant to translation and rotation if the co-ordinate system is appropriately chosen.

$$a_n = \frac{1}{L-1} \sum_{m=1}^{L-1} x_m e^{-i[2\pi/(L-1)]nm} \qquad b_n = \frac{1}{L-1} \sum_{m=1}^{L-1} y_m e^{-i[2\pi/(L-1)]nm}$$

- The coefficients a_n, b_n are not invariant, but after the transform

$$r_n = (|a_n|^2 + |b_n|^2)^{1/2}$$

  r_n are translation and rotation invariant.

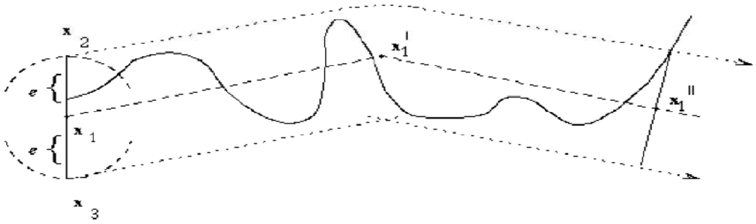
- To achieve magnification invariance the descriptors Wn are used: $W_n = r_n / r_1$
  The first 10-15 descriptors Wn are found to be sufficient for character description.

Boundary description using segment sequences

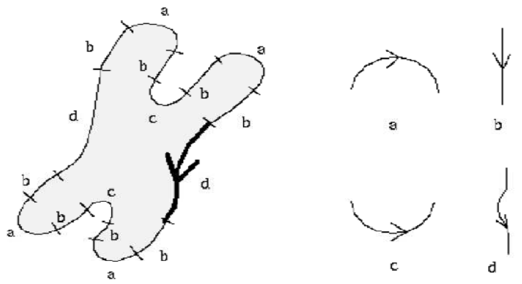
- If the segment type is known for all segments, the boundary can be described as a chain of segment types, a code-word consisting of representatives of a type alphabet.

- A polygonal representation approximates a region by a polygon, the region being represented using its vertices. Polygonal representations are obtained as a result of simple boundary segmentation.

- Another method for determining the boundary vertices is a tolerance interval approach based on setting a maximum allowed difference e.



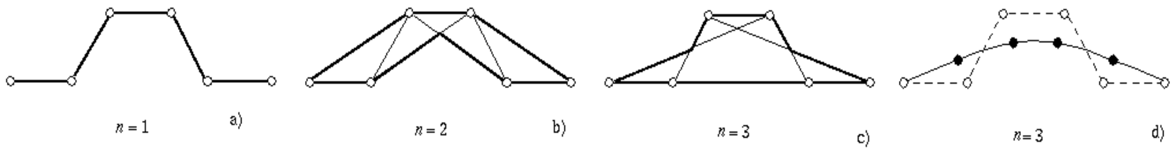
- Boundary segmentation into segments of constant curvature or curve segmentation into circular arcs and straight lines is used. Segments are considered as primitives for syntactic shape recognition procedures.



Structural description of chromosomes by a chain of boundary segments, code word: d,b,a,b,c,b,a,b,d,b,a,b,c,b,a,b.

B-spline representation

- Representation of curves using piecewise polynomial interpolation to obtain smooth curves is widely used in computer graphics.
- B-splines are piecewise polynomial curves whose shape is closely related to their control polygon - a chain of vertices giving a polygonal representation of a curve.
- B-splines of the third-order are most common because this is the lowest order which includes the change of curvature.
- Splines have very good representation properties and are easy to compute:
  - Firstly, they change their shape less then their control polygon, and do not oscillate between sampling points as many other representations do. A spline curve is always positioned inside a convex n+1-polygon for a B-spline of the n-th order.
  - Secondly, the interpolation is local in character. If a control polygon vertex changes its position, a resulting change of the spline curve will occur only in a small neighborhood of that vertex.
  - Thirdly, methods of matching region boundaries represented by splines to image data are based on a direct search of original image data.



Splines of order n. (a),(b),(c) Convex n+1 polygon for a B-spline of the nth order.
(d) 3rd order spline.

Q4. Explain region based shape representation and description

- A large group of shape description techniques is represented by heuristic approaches which yield acceptable results in description of simple shapes.
- Heuristic region descriptors: area, rectangularity, elongatedness, direction, compactness, etc.
- These descriptors cannot be used for region reconstruction and do not work for more complex shapes.
- Procedures based on region decomposition into smaller and simpler subregions must be applied to describe more complicated regions, and then sub regions can be described separately using heuristic approaches.

Q5. Discuss in detail the different scalar region descriptors.

Or Explain any three region based shape representation models.

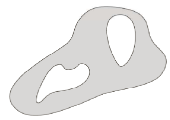
The different scalar region descriptors are explained below.

Area

- **Area** is given by the number of pixels of which the region consists.
- The *real* area of each pixel may be taken into consideration to get the *real* size of a region.
- If an image is represented as a rectangular raster, simple counting of region pixels will provide its area

Euler's number

- It is sometimes called **Genus** or the **Euler-Poincare characteristic**, describes a simple topologically invariant property of the object.
- **S** is the number of contiguous parts of an object and **N** is the number of holes in the object (an object can consist of more than one region).



$$\vartheta = S - N$$

Projections : Horizontal and vertical region projections

$$p_h(i) = \sum_j f(i,j)$$

$$p_v(j) = \sum_i f(i,j)$$

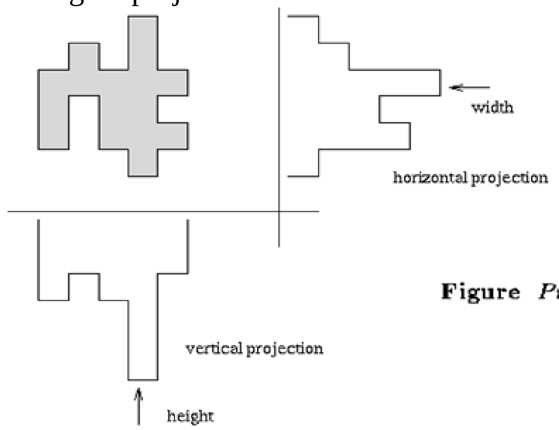


Figure *Projections.*

Eccentricity : The simplest is the ratio of major and minor axes of an object.
Elongatedness : A ratio between the length and width of the region bounding rectangle.

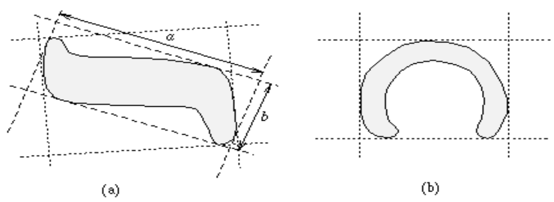


Figure 6.24 *Elongatedness: (a) Bounding rectangle gives acceptable results, (b) bounding rectangle cannot represent elongatedness.*

- Elongatedness can be evaluated as a ratio of the region area and the square of its thickness.
- The maximum region thickness (holes must be filled if present) can be determined as the number d of erosion steps that may be applied before the region totally disappears.

$$elongatedness = \frac{area}{(2d)^2}$$

Rectangularity

- Let $\mathbf{F_k}$ be the ratio of region area and the area of a bounding rectangle, the rectangle having the direction k.
- Rectangularity measured as a maximum of this ratio $\mathbf{F_k}$
- Elongatedness and rectangularity are independent of linear transformations -- translation, rotation, and scaling

$$rectangularity = \max_k(F_k)$$

Direction

- It is a property which makes sense in elongated regions only.
- If the region is elongated, direction is the direction of the longer side of a minimum bounding rectangle.
- If the shape moments are known, the direction \theta can be computed as

$$\theta = \frac{1}{2}tan^{-1}(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}})$$

- Direction is independent on all linear transformations which do not include rotation.
- Mutual direction of two rotating objects is rotation invariant.

Compactness

- Compactness is independent of linear transformations

$$compactness = \frac{(region\_border\_length)^2}{area}$$

- The most compact region in a Euclidean space is a circle.
- Compactness assumes values in the interval [1,infty) in digital images if the boundary is defined as an inner boundary,
- Independence from linear transformations is gained only if an outer boundary representation is used.

Q6. What are the factors need to be considered for a robust shape representation? Discuss any two region based approaches which address these factors.

The representation is said to be robust if it is invariant to transformations. The common transformations include translation, rotation and scaling. The representation is said to be invariant if it remains same even though the objects in the image are translated, rotated and scaled. Invariance to transformation is the major factor during the robust shape representation.

Moments

Region moment representations interpret a normalized gray level image function as a probability density of a 2D random variable. Properties of this random variable can be described using statistical characteristics - moments.

Assuming that non-zero pixel values represent regions, moments can be used for binary or gray level region description.

A moment of order (p+q) is dependent on scaling, translation, rotation, and even on gray level transformations and is given by

$$m_{pq} = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} i^p j^q f(i,j),$$

Where x, y, i, j are the region point co-ordinates (pixel co-ordinates in digitized images). **Translation invariance** can be achieved if we use the central moments.

$$\mu_{pq} = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} (i - x_c)^p (j - y_c)^q f(i,j), \qquad x_c = \frac{m_{10}}{m_{00}}, \qquad y_c = \frac{m_{01}}{m_{00}}$$

where x_c,y_c are the co-ordinates of the region's centroid.

Scale invariant features can also be found in scaled central moments η_{pq}

$$\eta_{pq} = \frac{\mu'_{pq}}{(\mu'_{00})^{\gamma}}, \qquad \gamma = \frac{p+q}{2} + 1, \qquad \mu'_{pq} = \frac{\mu_{pq}}{\alpha^{p+q+2}}$$

Rotation invariance can be achieved if the co-ordinate system is chosen such that μ_{11}=0.

Graph representation

Objects are represented by a planar graph with nodes representing sub regions resulting from region decomposition, and region shape is then described by the graph properties. There are two general approaches to acquiring a graph of sub regions:

- The first one is region thinning leading to the **region skeleton**, which can be described by a graph.
- The second option starts with the **region decomposition** into sub regions, which are then represented by nodes while arcs represent neighborhood relations of sub regions.

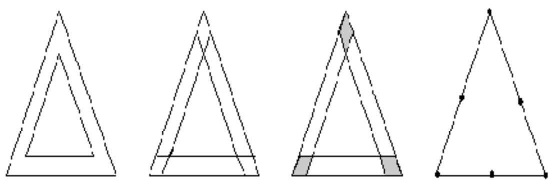
Graphical representation of regions has many advantages; the resulting graphs

- are **translation** and **rotation invariant**; position and rotation can be included in the graph definition
- are insensitive to small changes in shape are highly invariant with respect to region **magnitude (scale)**
- generate a representation which is understandable
- can easily be used to obtain the information-bearing features of the graph
- are suitable for syntactic recognition

Q7. Explain region decomposition with a neat diagram.

The decomposition approach is based on the idea that shape recognition is a hierarchical process. Shape primitives are defined at the lower level, primitives being thesimplest elements which form the region. A graph is constructed at the higher level - nodes result from primitives, arcs describe the mutual primitive relations.

Convex sets of pixels are one example of simple shape primitives.



Region decomposition
(a) Region. (b) Primary regions.
(c) Primary sub-regions and kernels.
(d) Decomposition graph.

The solution to the decomposition problem consists of two main steps:

- The first step is to segment a region into simpler sub regions (primitives)
- Second is the analysis of primitives.

Primitives are simple enough to be successfully described using simple scalar shape properties.

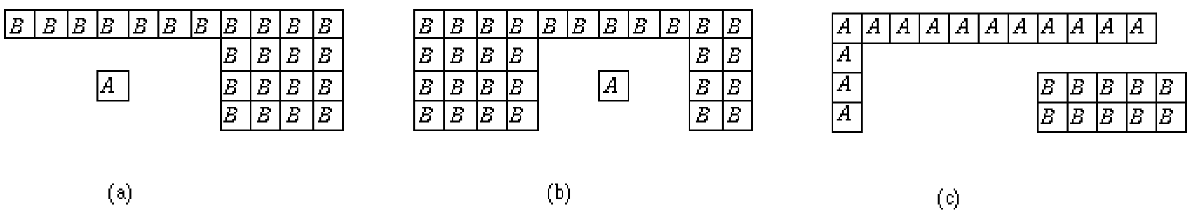
If sub regions are represented by polygons, graph nodes bear the following information;

1. **Node type** representing primary sub region or kernel.
2. **Number of vertices** of the sub region represented by the node.
3. **Area** of the sub region represented by the node.
4. **Main axis direction** of the sub region represented by the node.
5. **Center of gravity** of the sub region represented by the node.

If a graph is derived using attributes 1-4, the final description is **translation invariant**. A graph derived from attributes 1-3 is translation and **rotation invariant**. Derivation using the first two attributes results in a description which is **size invariant** in addition to possessing translation and rotation invariance.

Q8. Explain region neighborhood graphs

- Any time region decomposition into sub regions or image decomposition into regions is available, the region or image can be represented by a region neighborhood graph (the region adjacency graph being a special case).
- This graph represents every region as a graph node, and nodes of neighboring regions are connected by edges.
- A region neighborhood graph can be constructed from a quad tree image representation, from run-length encoded image data, etc.



Very often, the relative position of two regions can be used in the description process -- for example, a region A may be positioned to the **left of** a region B, or **above** B, or **close to** B, or a region C may **lie between** regions A and B, etc. We know the meaning of all of the given relations if A, B, C are points, but, with the exception of the relation **to be close**, they can become ambiguous if A, B, C are regions.

For instance, human observers are generally satisfied with the definition:

> The center of gravity of A must be positioned to the left of the leftmost point of B and (logical AND) the rightmost pixel of A must be left of the rightmost pixel of B

Q9. What are shape classes? Suggest any approach to identify the shape class of an object in an image.

The shape classes are expected to represent the **generic shapes** of the objects belonging to the class well and emphasize shape differences between classes, while the shape variations allowed within classes should not influence the description.

A widely used representation of in-class shape variations is determination of **class-specific** regions in the feature space. The feature space can be defined using a selection of shape features.

Another approach to shape class definition is to use a **single prototype shape** and determine a planar warping transform that if applied to the prototype produces shapes from the particular class. The prototype shape may be derived from examples.

If a set of **landmarks** can be identified on the regions belonging to specific shape classes, the landmarks can characterize the classes in a simple and powerful way. Landmarks are usually selected as easily recognizable border or region points.

*****