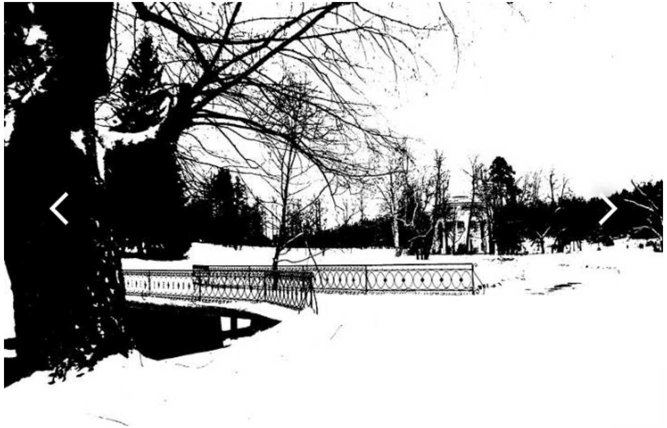
UNIT-3

Segmentation

THRESHOLDING:

•In digital image processing, thresholding is the simplest method of segmenting images. From a grayscale image, thresholding can be used to create binary images



DEFINITION

- The simplest thresholding methods replace each pixel in an image with a black pixel if the image intensity {\displaystyle I_{i,j}}I_{{i,j}} is less than some fixed constant T (that is, {\displaystyle I_{i,j}<T}I_{{i,j}}<T), or a white pixel if the image intensity is greater than that constant. In the example image on the right, this results in the dark tree becoming completely black, and the white snow becoming completely white.

CATEGORIES OF THRESHOLDING:

- Histogram shape-based methods, where, for example, the peaks, valleys and curvatures of the smoothed histogram are analyzed
- Clustering-based methods, where the gray-level samples are clustered in two parts as background and foreground (object), or alternately are modeled as a mixture of two Gaussians
- Entropy-based methods result in algorithms that use the entropy of the foreground and background regions, the cross-entropy between the original and binarized image, etc.
- Object Attribute-based methods search a measure of similarity between the gray-level and the binarized images, such as fuzzy shape similarity, edge coincidence, etc.

- Local methods adapt the threshold value on each pixel to the local image characteristics. In these methods, a different T is selected for each pixel in the image.
- Spatial methods [that] use higher-order probability distribution and/or correlation between pixels

EDGE-BASED SEGMENTATION

- Edge-based segmentation represents a large group of methods based on information about edges in the image
- Edge-based segmentations rely on edges found in an image by edge detecting operators -- these edges mark image locations of discontinuities in gray level, color, texture, etc.
- Image resulting from edge detection cannot be used as a segmentation result.
- Supplementary processing steps must follow to combine edges into edge chains that correspond better with borders in the image.

- The final aim is to reach at least a partial segmentation -- that is, to group local edges into an image where only edge chains with a correspondence to existing objects or image parts are present.
- The more prior information that is available to the segmentation process, the better the segmentation results that can be obtained.
- The most common problems of edge-based segmentation are
  - ✓an edge presence in locations where there is no border, and
  - ✓no edge presence where a real border exists.

EDGE IMAGE THRESHOLDING

- Almost no zero-value pixels are present in an edge image, but small edge values correspond to non-significant gray level changes resulting from quantization noise, small lighting irregularities, etc.
- Selection of an appropriate global threshold is often difficult and sometimes impossible; p-tile thresholding can be applied to define a threshold
- Alternatively, non-maximal suppression and hysteresis thresholding can be used as was introduced in the Canny edge detector.

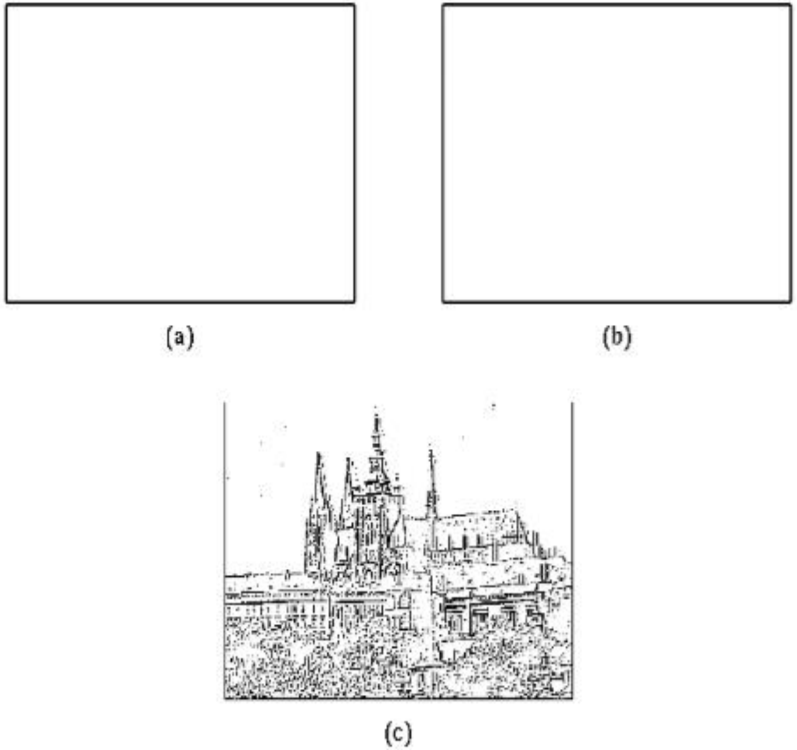


Figure 5.9 *Edge image thresholding: (a) Original image, (b) edge image, (c) thresholded edge image.*

EDGE RELAXATION

- Borders resulting from the previous method are strongly affected by image noise, often with important parts missing.
- Considering edge properties in the context of their mutual neighbors can increase the quality of the resulting image.

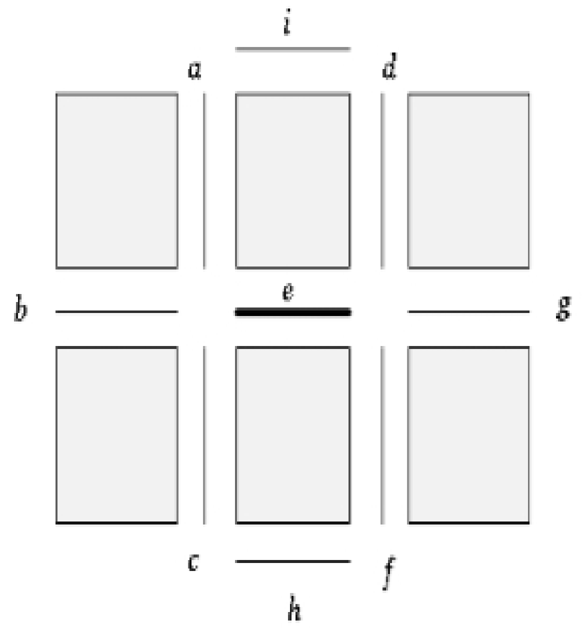
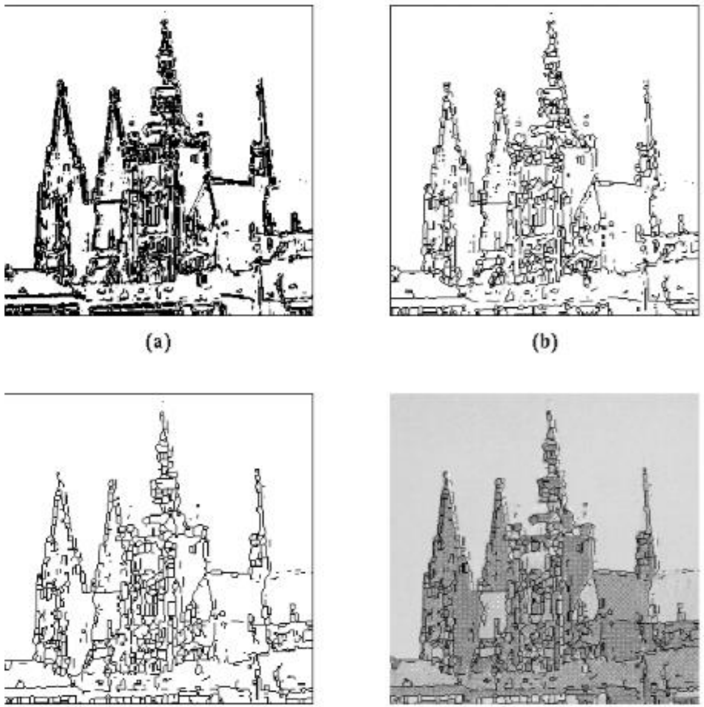


Figure 5.10 *Crack edges surrounding central edge e.*

- Edge relaxation is an iterative method, with edge confidences converging either to zero (edge termination) or one (the edge forms a border).
- The confidence of each edge e in the first iteration can be defined as a normalized magnitude of the crack edge,
- with normalization based on either the global maximum of crack edges in the whole image, or on a local maximum in some large neighborhood of the edge



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

Figure 5.12 *Edge relaxation, see Figure 5.9a for original: (a) Resulting borders after 10 iterations, (b) borders after thinning, (c) borders after 100 iterations, thinned (d) borders after 100 iterations overlaid over original.*

REGION BASED SEGMENTATION

- Region based segmentation is a technique For dertermining region
- Directly.

- region growing is a technique simple region based image segmentation method.
- it is a classified as pixel based image
- segmentation method.

APPLICATION AND METHOD

- Application finding famous,veins etc in medical image,finding targets in satelite/aerial image,finding people in surveillance image,summarazing video etc.
- Method theresholldding,k-means clustering etc.

FORMULATION

- Complete the segmentation must be complete Ui=1,Ri=R Every pixel must be in a region.
- Connectedness: The points of a region must be connected in some sense.
- Disjoint region must be disjoint Ri n Rj.

FORMULATION

- Satistiability pixel of a region must satisfy one common p(Ri)=TRUE,property for fast.
- Segmentation ability different regien satisfy
- Different p(RiuR)=FALSE properties.

REGION GROWING

- Region growing is a procedure that groups pixels or Sub regien into large regien.
- The simplest of these approach is pixel aggregation which starts with a set of speed points and from these grows regien by appending to each points of these neighbouring pixels that have similar properties

ADVANTAGES OF REGIEN GROWING

- Regien growing Method can provide the original image which have clear edges good segmentation results.
- The concepts in simple we only need a small number of speed points to represent the property.

DISADVANTAGES REGIEN GROWING

- Completationally expensive.
- It is a local method with no global of the problem senstitive noise.

REGIEN SPLITTING

- Region growing from set of seed points.
- An alternate is to start with whole image as a single regien and subdivide the regiens do not satisfy condition of homogeneity.

REGIEN MEMBERRING:

- Regien merging of the opposite of regien splitting.
- Start with small regien.
- Typically,spiltting and merging approaches are called iteratively.

MATCHING

- The best match is based on some criterion of optimality which depends on object properties and object relations.
- Matching is another basic approach to segmentation that can be used to locate known objects in an image, to search for specific patterns, etc.

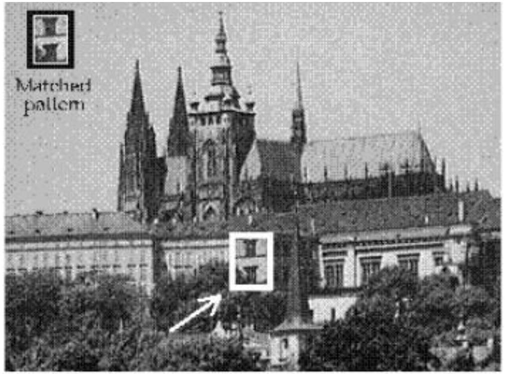


Figure 5.53 *Segmentation by matching; matched pattern and location of the best match.*

- Matched patterns can be very small, or they can represent whole objects of interest.
- While matching is often based on directly comparing gray-level properties of image subregions, it can be equally well performed using image-derived features or higher-level image descriptors.
- In such cases, the matching may become invariant to image transforms.
- Criteria of optimality can compute anything from simple correlations up to complex approaches of graph matching

MATCHING CRITERIA

- Exact copy of the pattern of interest cannot be expected in the processed image - some part of the pattern is usually corrupted in real images by noise, geometric distortion, occlusion, etc.
- Search for locations of maximum match is appropriate.

- Matching criteria can be defined in many ways; in particular, correlation between a pattern and the searched image data is a general matching criterion.

Algorithm 5.24: Match-based segmentation

1. Evaluate a match criterion for each location and rotation of the pattern in the image.
2. Local maxima of this criterion exceeding a preset threshold represent pattern locations in the image.

•Possible matching optimality criteria describing a match between f and h located at a position (u,v):

•The "1" added to each denomiator to prevents dividing by zero for a perfect match.

•The cost is evaluated at each (u,v) pixel location in the image being processed.

•Possible implementation decissions include whether the pattern is only computed entirely within the image or partial pattern positions when the pattern crosses image borders

$$C_1(u,v) = \frac{1}{\max_{(i,j)\in V} \left|f(i+u, j+v) - h(i,j)\right| + 1}$$

$$C_2(u,v) = \frac{1}{\sum_{(i,j)\in V} \left|f(i+u, j+v) - h(i,j)\right| + 1}$$

$$C_3(u,v) = \frac{1}{\sum_{(i,j)\in V} \left[f(i+u, j+v) - h(i,j)\right]^2 + 1}$$

- A simple example of the C_3 optimality criterion values is given:

$$
\begin{vmatrix}
1 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 8
\end{vmatrix}
\qquad
\begin{vmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1
\end{vmatrix}
\qquad
\begin{vmatrix}
\underline{\mathbf{1/3}} & 1/6 & 1/8 & \times & \times \\
1/5 & 1/7 & 1/8 & \times & \times \\
1/8 & 1/9 & 1/57 & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times
\end{vmatrix}
$$

(a) (b) (c)

Figure 5.54 *Optimality matching criterion evaluation: (a) Image data, (b) matched pattern, (c) optimality criterion values, the best match underlined.*

- If a fast, effective Fourier transform algorithm is available, the convolution theorem can be used to evaluate matching.

- The correlation between a pattern h and image f can be determined by first taking the product of the Fourier transform F of the image f and the complex conjugate of the Fourier transform H^# of the pattern h and then applying the inverse transform.
- To compute the product of Fourier transforms, F and H^# must be of the same size; if a pattern size is smaller, zero-valued lines and columns can be added to inflate it to the appropriate size.
- Sometimes, it may be better to add non-zero numbers, for example, the average gray level of processed images can serve the purpose well.

CONTROL STRATEGIES OF MATCHING

- These copies must match the pattern in size and rotation, and the geometric distortion must be small.
- To adapt match-based methods to detect patterns that are rotated, enlarged, and/or reduced, it would be necessary to consider patterns of all possible sizes and rotations.
- Match-based segmentation localizes all image positions at which close copies of the searched pattern are located.
- Another option is to use just one pattern and match an image with all possible geometric transforms of this pattern, and this may work well if some information about the probable geometric distortion is available.
- Note that there is no difference in principle between these approaches.

- Matching can be used even if an infinite number of transformations are allowed. Let us suppose a pattern consists of parts, these parts being connected by rubber links.
- Even if a complete match of the whole pattern within an image may be impossible, good matches can often be found between pattern parts and image parts.
- Good matching locations may not be found in the correct relative positions, and to achieve a better match, the rubber connections between pattern parts must be either pushed or pulled.
- The final goal can be described as the search for good partial matches of pattern parts in locations that cause minimum force in rubber link connections between these parts.
- A good strategy is to look for the best partial matches first, followed by a heuristic graph construction of the best combination of these partial matches in which graph nodes represent pattern parts.

- The sequence of match tests must be data driven.
- Fast testing of image locations with a high probability of match may be the first step, then it is not necessary to test all possible pattern locations.
- Another speed improvement can be realized if a mismatch can be detected before all the corresponding pixels have been tested.
- The correlation changes slowly around the best matching location ... matching can be tested at lower resolution first, looking for an exact match in the neighborhood of good low-resolution matches only.

ACTIVE CONTOUR MODELS

- Active contour is a type of segmentation technique which can be defined as use of energy forces and constraints for segregation of the pixels of interest from the image for further processing and analysis. Active contour described as active model for the process of segmentation.
- Contours are boundaries designed for the area of interest required in an image.
- Contour is a collection of points that undergoes interpolation process. The interpolation process can be linear, splines and polynomial which describes the curve in the image

- Different models of active contours are applied for the segmentation technique in image processing. The main application of active contours in image processing is to define smooth shape in the image and forms closed contour for the region.
- Active contour models involve snake model, gradient vector flow snake model, balloon model and geometric or geodesic contours.
- Active contours can be defined as the process to obtain deformable models or structures with constraints and forces in an image for segmentation.

- For the set of points in an image, the contour can be defined based on forces and constraints in the regions of the image. Active contours are used in various applications in the segmentation of the medical images
- Different types of active contour models are used in various medical applications especially for the separation of required regions from the various medical images.
- For example, a slice of brain CT image is considered for segmentation using active contour models.
- The contour of the image defines the layers of the region in the brain which is shown in the

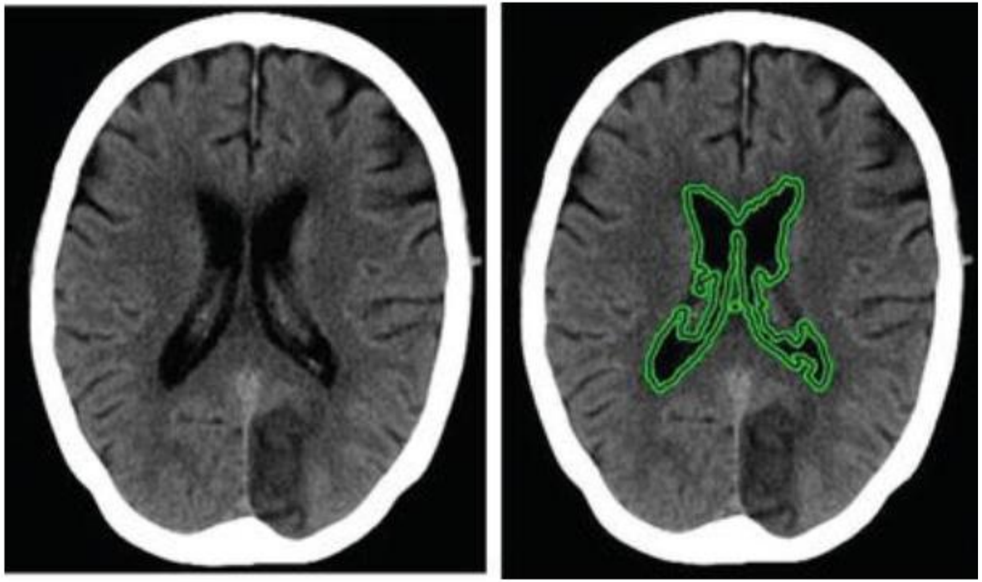
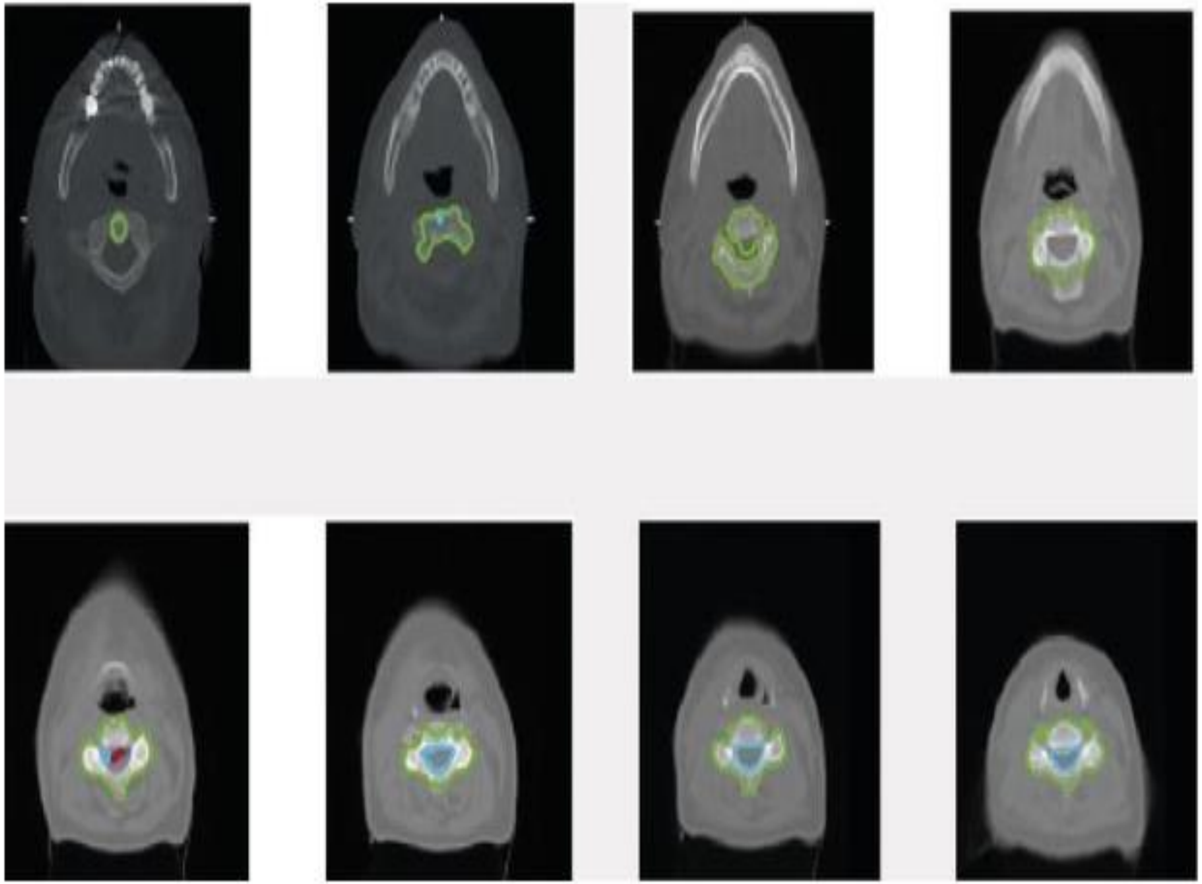


Figure 1.
Segmentation of brain CT image using active contours.

- Active contours can also be used for segmentation of 3-D images derived from different medical imaging modalities.
- 2-D slices of image data are used for the separation of target object from the 3-D images.
- These 2-D slices of images in all directions along with the segmented target region are subjected to 3-D reconstruction to segregate the regions
- The 2-D slice of the head CT image and mesh model designed for that 2-D slice of head CT image in all directions is defined in Figure 2(a) and (b) respectively.

- The iterative segmentation of each and every region of the thoracic cavity from the head CT image is shown

Figure 3.
Iterative 3-D segmentation of head CT image using active contours.



SNAKE MODEL

- Snake model is a technique that has the potential of solving wide class of segmentation cases. The model mainly works to identify and outlines the target object considered for segmentation.
- Active snake model also called snakes generally configures by the application of spline focussed to minimise energy followed by various forces governing the image.
- Snake model enacts deformable model to an image through energy minimisation.
- Snake works efficiently with complex target objects by breaking down the figure into various smaller targets

- Snake model that has more advantages than utilising implicit and explicit curve forms
- $v(s,t)=(x(s,t),y(s,t))$E1
- where x and y are the coordinates of the two-dimensional curve, v is spline parameter in the range 0–1, s is linear parameter $\in [0,1]$ and t is time parameter $\in [0, \infty]$.

- The forces in snake include external forces as well as image forces that helps in feature identification.

- Contour is developed around the left lung which can be used for further processing.

Segmentation of chest image using snake model.

- The traditional method of active snake model has several inefficiencies like insensitivity to noises, false contour detection in high complex objects which are solved in advanced versions of contour methods.

FUZZY CONNECTIVITY

- Fuzzy Connected (FC) Image Segmentation
- • FC has been used with considerable success in medical (and
- other) images.
- – Udupa and Samarasekera were the first to use FC in medical images.
- (Graphical Models and Image Processing, 1996)

FC FAMILY

- Absolute FC
- Scale-based FC (b-, t-, g-scale based)
- Relative FC
- Iterative Relative FC
- Vectorial FC
- Hierarchical FC
- Model-based FC

- Defining the shape of an object can prove to be very difficult. Shape is usually represented verbally or in figures.
- There is no generally accepted methodology of shape description. Further, it is not known what in shape is important.
- Current approaches have both positive and negative attributes; computer graphics or mathematics use effective shape representations which are unusable in shape recognition and vice versa.
- In spite of this, it is possible to find features common to most shape description approaches.

COMMON SHAPE DESCRIPTION METHODS CAN BE CHARACTERIZED FROM DIFFERENT POINTS OF VIEW

• rotation, and scale transformations: Shape description proInput representation form: Object description can be based on boundaries or on more complex knowledge of whole regions

•Object reconstruction ability: That is, whether an object's shape can or cannot be reconstructed from the description.

•Incomplete shape recognition ability: That is, to what extent an object's shape can be recognized from the description if objects are occluded and only partial shape information is available.

- Local/global description character: Global descriptors can only be used if complete object data are available for analysis. Local descriptors describe local object properties using partial information about the objects. Thus, local descriptors can be used for description of occluded objects.

- Mathematical and heuristic techniques: A typical mathematical technique is shape description based on the Fourier transform. A representative heuristic method may be elongatedness.
- Statistical or syntactic object description.
- A robustness of description to translationperties in different resolutions.

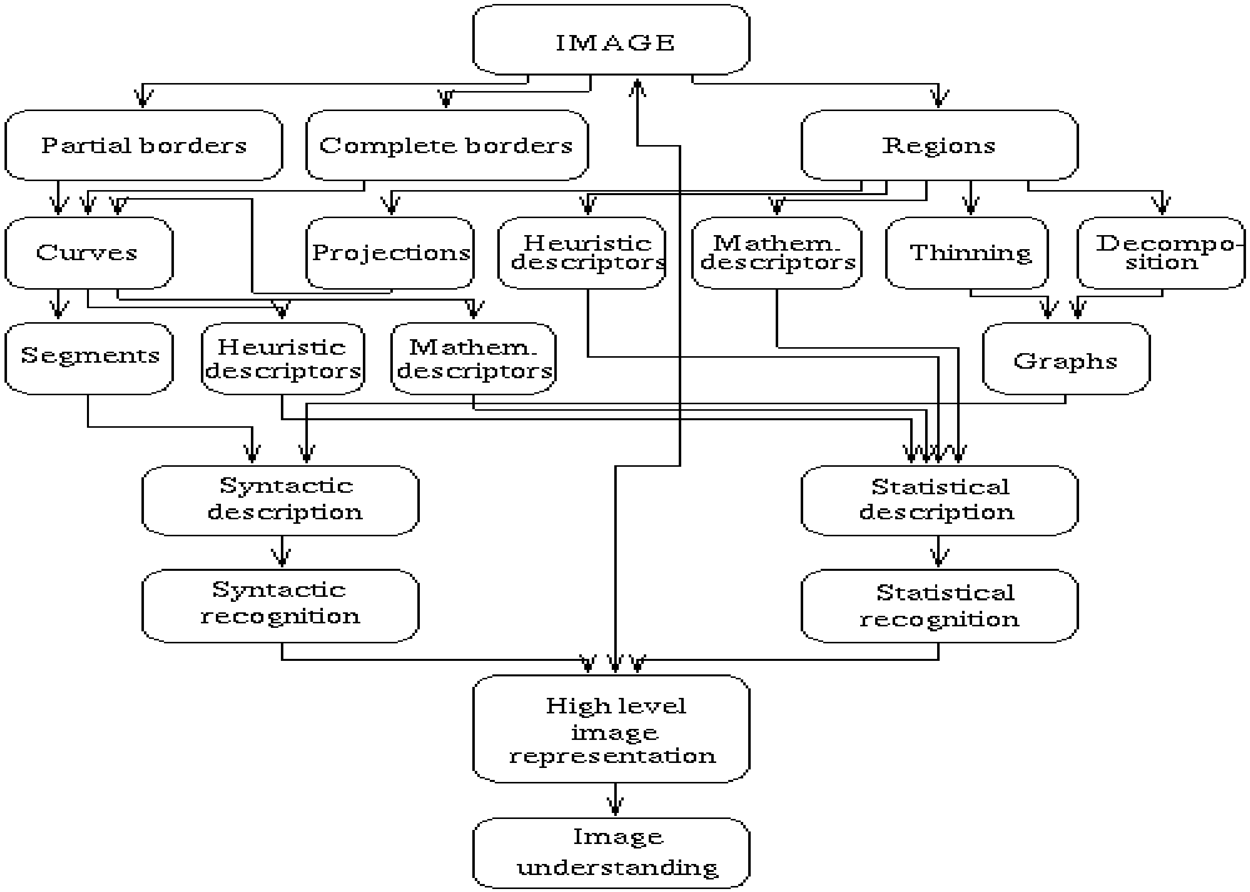


Figure 6.1 *Image analysis and understanding methods.*

- Sensitivity to scale is even more serious if a shape description is derived, because shape may change substantially with image resolution.
- Therefore, shape has been studied in multiple resolutions which again causes difficulties with matching corresponding shape representations from different resolutions.
- Moreover, the conventional shape descriptions change discontinuously.
- A scale-space approach aims to obtain continuous shape descriptions if the resolution changes continuously.

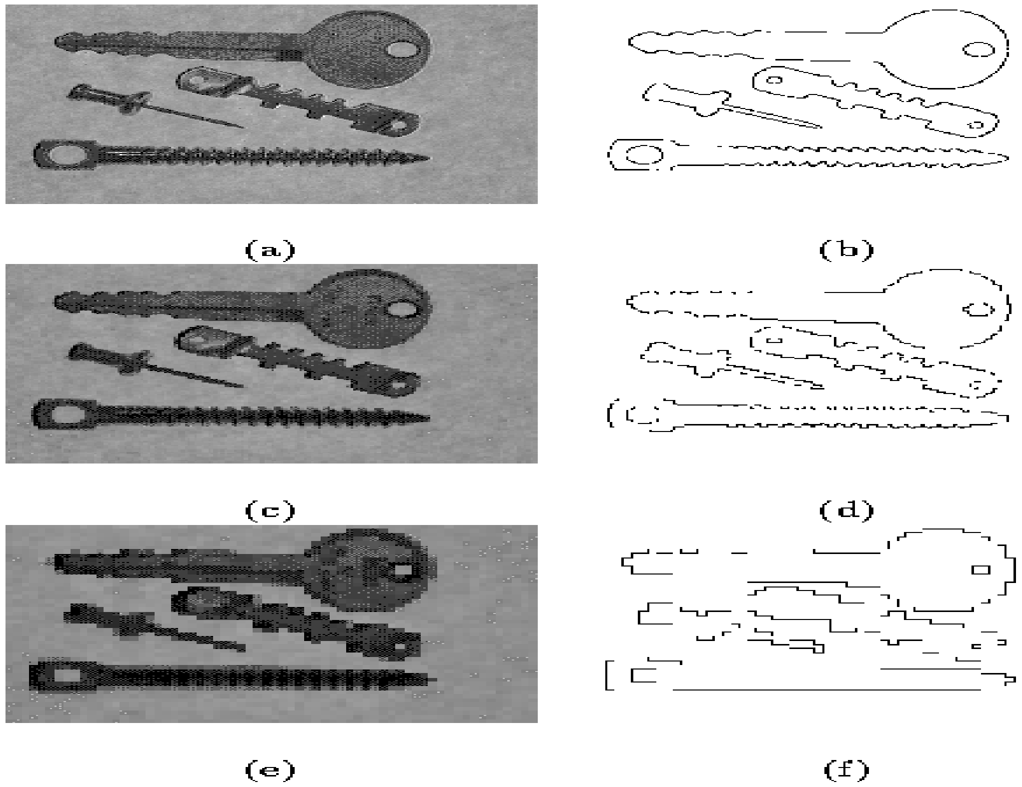


Figure 6.2 *(a) Original image 640 × 480, (b) contours of a, (c) original image 160 × 120 (d) contours of c, (e) original image 64 × 48, (f) contours of e.*

- In many tasks, it is important to represent classes of shapes properly, e.g. shape classes of apples, oranges, pears, bananas, etc.
- The **shape classes** should represent the generic shapes of the objects belonging to the same classes well. Obviously, shape classes should emphasize shape differences among classes while the influence of shape variations within classes should not be reflected in the class description.

REGION IDENTIFICATION

- Region identification is necessary for region description. One of the many methods for region identification is to label each region (or each boundary) with a unique (integer) number; such identification is called **labeling** or **coloring**, also **connected component labeling.**
- Goal of segmentation was to achieve complete segmentation, now, the regions must be labeled.
- $$R_b^C = \bigcup_{i=1, i \neq b}^{m} R_i$$

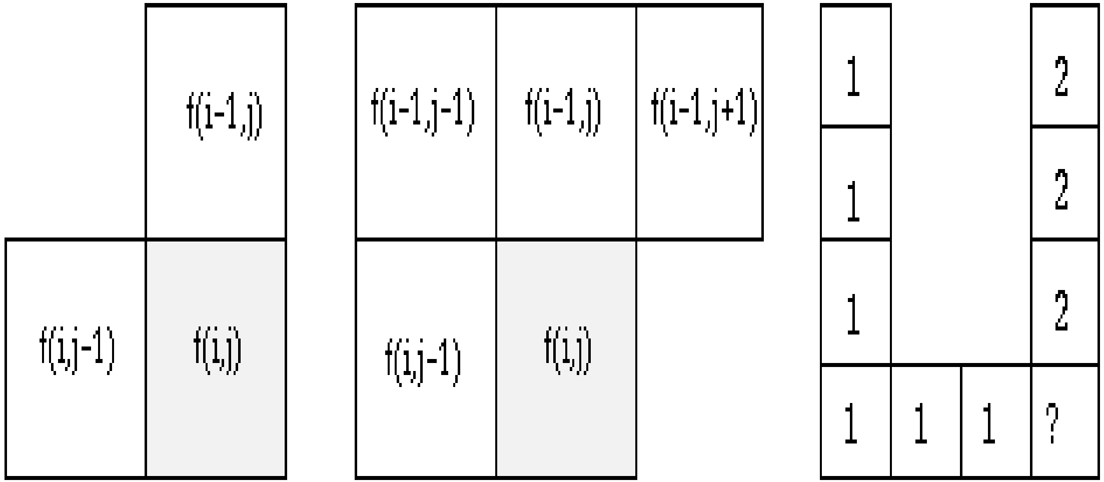


Figure 6.3 *Masks for region identification: (a) In 4-connectivity, (b) in 8-connectivity, (c) label collision.*

- **Label collision** is a very common occurrence -- examples of image shapes experiencing this are U-shaped objects, mirrored E objects, etc.
- The equivalence table is a list of all label pairs present in an image; all equivalent labels are replaced by a unique label in the second step.
- The algorithm is basically the same in 4-connectivity and 8-connectivity, the only difference being in the neighborhood mask shape.
-

INTRODUCTION:

- More and more images have been generated in digital form around the world. There is a growing interest in finding images in large collections or from remote databases. In order to find an image, the image has to be described or represented by certain features. Shape is an important visual feature of an image. Searching for images using shape features has attracted much attention. There are many shape representation and description techniques in the literature. In this paper, we classify and review these important techniques. We examine implementation procedures for each technique and discuss its advantages and disadvantages. Some recent research results are also included and discussed in this paper. Finally, we identify some promising techniques for image retrieval according to standard principles.

- Abstract

- More and more images have been generated in digital form around the world. There is a growing interest in finding images in large collections or from remote databases. In order to find an image, the image has to be described or represented by certain features. Shape is an important visual feature of an image. Searching for images using shape features has attracted much attention. There are many shape representation and description techniques in the literature. In this paper, we classify and review these important techniques. We examine implementation procedures for each technique and discuss its advantages and disadvantages. Some recent research results are also included and discussed in this paper. Finally, we identify some promising techniques for image retrieval according to standard principles.

A CONTOUR-BASED APPROACH TO BINARY SHAPE CODING USING A MULTIPLE GRID CHAIN CODE

- This paper presents a contour-based approach to efficiently code binary shape information in the context of object-based video coding.
- This approach meets some of the most important requirements identified for the MPEG-4 standard, notably efficient coding and low delay.
- The proposed methods support both object-based lossless and quasi-lossless coding modes.
- For the cases where low delay is a primary requirement, a macroblock-based coding mode is proposed which can take advantage of inter-frame coding to improve the coding efficiency …

- **Chain codes**
- Chain codes describe an object by a sequence of unit-size line segments with a given orientation.
- The first element of such a sequence must bear information about its position to permit the region to be reconstructed.
- 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 if the description chain is interpreted as a base four number -- that pixel is then used as the starting pixel.
  - A *mod* 4 or *mod* 8 difference is called a chain code **derivative**.
-

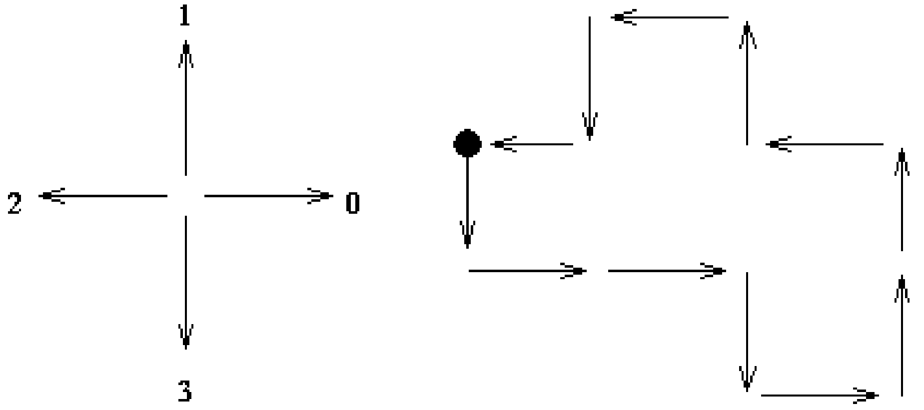


Figure 6.6 *Chain code in 4-connectivity, and its derivative. Code: 3, 0, 0, 3, 0, 1, 1, 2, 1, 2, 3, 2, derivative: 1, 0, 3, 1, 1, 0, 1, 3, 1, 1, 3, 1.*

- **Simple geometric border representation**
- The following descriptors are mostly based on geometric properties of described regions. Because of the discrete character of digital images, all of them are sensitive to image resolution.
    - *Boundary length*
    - *Curvature*

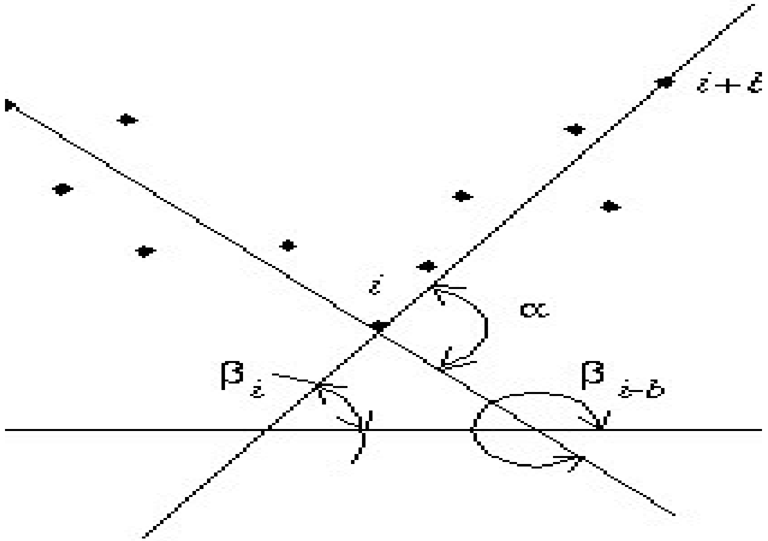
SIMPLE GEOMETRIC BORDER REPRESENTATION

simple geromatric border representation are based on geomatirc properties of described regions e. g:

- ✓ Boundary length
- ✓ Curvature
- ✓ Bending energy
- ✓ Signature
- ✓ Chord distribution

Because of the discrete character of digital images, all of them are sensitive to image resolution.

- **Boundary Length**
- **Curvature:**



Bending Energy

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

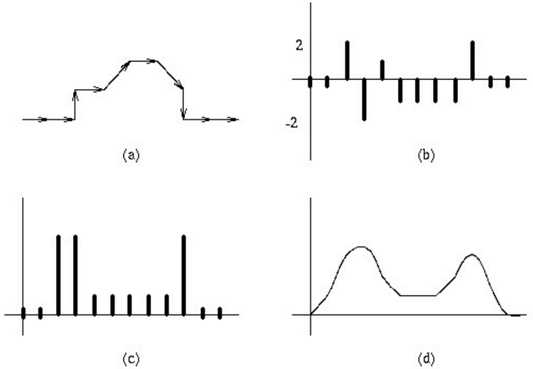


Figure 6.8 *Bending energy: (a) Chain code 0, 0, 2, 0, 1, 0, 7, 6, 0, 0, (b) curvature 0, 2, -2, 1, -1, -1, -1, 2, 0, (c) sum of squares gives the bending energy, (d) smoothed version.*

- Signature:

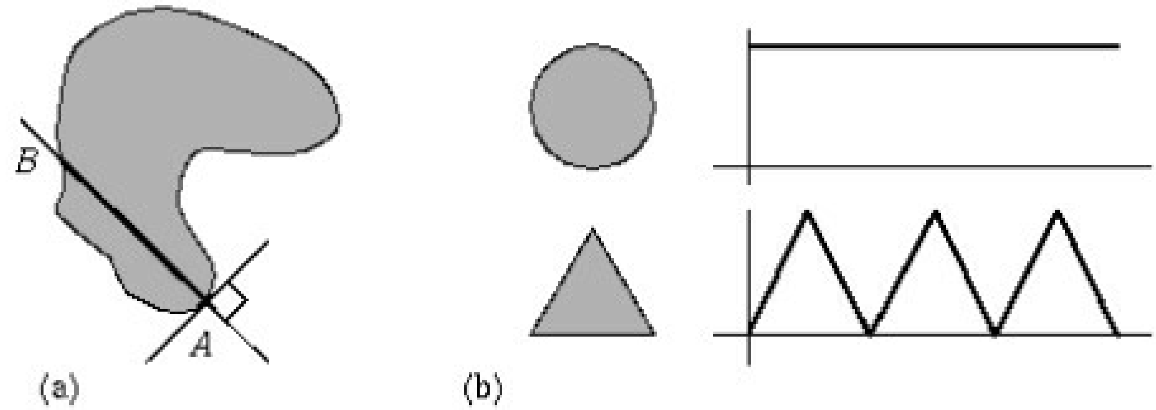


Figure 6.9 *Signature: (a) Construction, (b) signatures for a circle and a triangle.*

- Chord is a line joining any two points of the region boundary is a chord.
- ○**Chord:** Let b(x,y)=1 repersent the contour points, and b(x,y)=0 represent all other points.

$$h(\Delta x, \Delta y) = \int\int b(x,y)b(x+\Delta x, y+\Delta y)dx\,dy$$

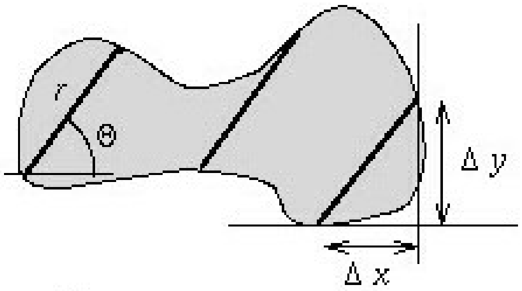
$$h(\Delta x, \Delta y) = \sum_i \sum_j b(i,j)b(i+\Delta x, j+\Delta y)$$

✓**Rotation** - independent radial distribution.

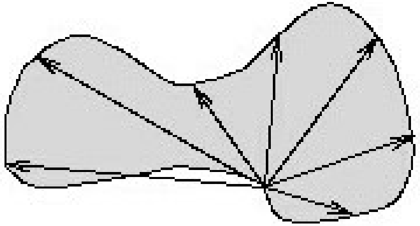
$$h_r(r) = \int_{-\pi/2}^{\pi/2} h(\Delta x, \Delta y) r\,d\theta$$

- The angular distribution h_a(theta)is independent of scale, while rotation cause a propertional off set.

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



(a)



(b)

REGION BASED SHAPE REPRESENTATION&DESCRIPTION

Simple scalar region 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, then subregions can be described separately using heuristic approaches.
- Simple scalar region descriptors
- 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.
- If the image is represented by a quadtree, then:

Algorithm 6.4: Calculating area in quadtrees

1. Set all region area variables to zero, and determine the global quadtree depth H; for example, the global quadtree depth is $H = 8$ for a 256×256 image.

2. Search the tree in a systematic way. If a leaf node at a depth h has a non-zero label, proceed to step (3).

- The region can also be represented by n polygon vertices

$$area = \frac{1}{2}\left|\sum_{k=0}^{n-1}(x_k y_{k+1} - x_{k+1} y_k)\right|$$

the sign of the sum represents the polygon orientation.

•If the region is represented by the (anti-clockwise) Freeman chain code the following algorithm provides the area

Algorithm 6.5: Region area calculation from Freeman 4-connectivity chain code representation

1. Set the region *area* to zero. Assign the value of the starting point *i* co-ordinate to the variable *vertical_position*.

2. For each element of the chain code (values 0, 1, 2, 3) do

```
switch(code) {
     case 0:
          area := area - vertical_position;
          break;
     case 1:
          vertical_position := vertical_position + 1;
          break;
     case 2:
          area := area + vertical_position;
          break;
     case 3:
          vertical_position := vertical_position - 1;
          break;
}
```